Les Cahiers du GERAD

ISSN: 0711–2440

Comptes rendus du neuvième atelier de résolution de problèmes industriels de Montréal, 19-23 août 2019

Proceedings of the ninth Montréal industrial problem solving workshop, August 19-23, 2019

Odile Marcotte, Editor

G–2020–57

October 2020

La publication de ces rapports de recherche est rendue possible grâce au soutien de HEC Montréal, Polytechnique Montréal, Université McGill, Université du Québec à Montréal, ainsi que du Fonds de recherche du Québec – Nature et technologies.

Dépôt légal – Bibliothèque et Archives nationales du Québec, 2020
– Bibliothèque et Archives Canada, 2020

The publication of these research reports is made possible thanks to the support of HEC Montréal, Polytechnique Montréal, McGill University, Université du Québec à Montréal, as well as the Fonds de recherche du Québec – Nature et technologies.

Legal deposit – Bibliothèque et Archives nationales du Québec, 2020
– Library and Archives Canada, 2020

GERAD HEC Montréal
3000, chemin de la Côte-Sainte-Catherine
Montréal (Québec) Canada H3T 2A7

Tél. : 514 340-6053
Téléc. : 514 340-5665
info@gerad.ca
www.gerad.ca

Préface

Le Neuvième atelier de résolution de problèmes industriels de Montréal, qui eut lieu du 19 au 23 août 2019, fut organisé conjointement par le CRM et l'Institut de valorisation des données (IVADO). Il attira plus de 100 participants et permit à neuf équipes d'examiner des problèmes fournis par le Conseil national de recherches du Canada (CNRC) et cinq compagnies ou institutions : Radio-Canada, l'Autorité des marchés financiers (AMF), Air Canada, Desjardins et Co-operators. Quatre de ces compagnies ou institutions étaient déjà des partenaires d'IVADO : seule la compagnie Co-operators ne l'était pas mais elle apporta une contribution financière à l'organisation de l'atelier. Le CNRC et l'Institut canadien des sciences statistiques firent aussi une contribution à l'atelier.

Les problèmes provenaient de domaines variés et requéraient des expertises diverses. Les problèmes soumis par Radio-Canada et l'AMF relevaient du traitement de la langue naturelle : Philippe Langlais (U. de Montréal) et Jian-Yun Nie (U. de Montréal) coordonnaient respectivement les équipes qui les ont examinés. Air Canada fournit trois problèmes à l'atelier : deux problèmes reliés à la gestion du revenu et un autre sur l'optimisation d'un programme de fidélité (Aeroplan). Les équipes étudiant les trois problèmes d'Air Canada avaient pour coordonnateurs respectifs Fabian Bastin (U. de Montréal), François Bellavance (HEC Montréal) et Margarida Carvalho (U. de Montréal).

La compagnie Desjardins soumit un problème de segmentation et lissage de territoires dont les coordonnateurs étaient Philippe Gagnon (Oxford et U. de Montréal) et Juliana Schulz (HEC Montréal), et Co-operators un problème sur la détection de fraudes dont le coordonnateur était Anas Abdallah (McMaster). Le CNRC fournit un problème de représentation de structure requérant une expertise en intelligence artificielle : Guy Wolf (U. de Montréal) était le coordonnateur de ce problème. Le deuxième problème du CNRC fut soumis par des astronomes et portait sur la détection de radio-fréquences dans le cadre de la surveillance d'un site : le professeur Chris Budd (Bath) coordonnait les travaux de l'équipe examinant ce problème.

Je remercie chaleureusement nos partenaires industriels et institutionnels, les coordonnateurs des équipes, ainsi que les responsables de la rédaction des rapports inclus dans les comptes rendus, en particulier David Alfonso Hermelo, Pan Du, Gabriel Lemyre et Adel Nabli (de l'Université de Montréal), Mohammad Daneshvar et Francis Duval (de l'Université du Québec à Montréal), Michael Lindstrom (de UCLA), Scott Gigante (de l'Université Yale) et Chris Budd (de l'Université de Bath). Le succès de l'atelier est le fruit de leurs efforts et de leur enthousiasme! Finalement j'exprime toute ma reconnaissance à Karine Hébert, qui m'a aidée à mettre en forme ces comptes rendus.

Odile Marcotte
Professeure associée, UQAM
Membre associé, GERAD

CENTRE DE RECHERCHES MATHÉMATIQUES

IVADO

Foreword

The Ninth Montreal IPSW took place on August 19-23, 2019, and was jointly organized by the CRM and IVADO (Institute for Data Valorization). The workshop welcomed more than 100 participants and allowed nine teams to study problems submitted by the National Research Council of Canada (NRC) and five companies or institutions: Radio-Canada (the French network of the CBC), the Autorité des marchés financiers (AMF) of the Québec Government, Air Canada, Desjardins, and The Co-operators. Four of these companies were already IVADO partners: The Co-operators was not an IVADO partner but made a financial contribution to the workshop, as did the NRC and CANSSI (the Canadian Statistical Sciences Institute).

The problems submitted to the workshop were varied and required expertise from diverse fields. Those submitted by Radio-Canada and the AMF required expertise in NLP (Natural Language Processing): Philippe Langlais (U. de Montréal) and Jian-Yun Nie (U. de Montréal) were the respective coordinators of the teams studying these problems. Air Canada provided the workshop with three problems, i.e., two problems in revenue management and a third on the optimization of a loyalty program (Aeroplan). The three corresponding teams were led (respectively) by Fabian Bastin (U. de Montréal), François Bellavance (HEC Montréal), and Margarida Carvalho (U. de Montréal).

An insurance group within Desjardins submitted a problem on the geographic stratification of risk, whose coordinators were Philippe Gagnon (Oxford and U. de Montréal) and Juliana Schulz (HEC Montréal). The Co-operators submitted a problem on fraud detection: its coordinator was Anas Abdallah (McMaster). The NRC submitted two problems: one on structure representation, whose team was led by Guy Wolf (U. de Montréal), a researcher in artificial intelligence; and another on the unsupervised learning of novel RFI sources. The latter problem was submitted by a group of astronomers and was studied by a team led by Professor Chris Budd (Bath).

I extend my warmest thanks to our industrial and institutional partners, to the team coordinators, and to the persons responsible for the reports found in the proceedings, in particular: David Alfonso Hermelo, Pan Du, Gabriel Lemyre, and Adel Nabli (Université de Montréal); Mohammad Daneshvar and Francis Duval (Université du Québec à Montréal); Michael Lindstrom (UCLA); Scott Gigante (Yale); and Chris Budd (Bath). The workshop was successful because of their contributions and enthusiasm! I am also very grateful to Karine Hébert, who helped me put these proceedings together.

Odile Marcotte
Adjunct Professor, UQAM
Associate member, GERAD

CENTRE
DE RECHERCHES
MATHÉMATIQUES

IVADO

Contents

David Alfonso et al.
1 Content classification and keyword extraction for `ici.radio-canada.ca` 6

Pan Du et al.
2 Role-oriented legal information retrieval 15

Caroline Dietrich et al.
3 Constrained demand – Air Canada 30

Ismael Assani et al.
4 Flight spill detection 37

Federico Bobbio et al.
5 Optimizing the design of a loyalty program 51

Delphine Boursicot et al.
6 Poisson regression for smooth geographic stratification of risk. 64

Anas Abdallah et al.
7 Prioritizing abnormal situations automatically in the context of fraudulent claims 74

David Bernier et al.
8 Development of a mathematical framework to represent a 2D/3D smooth geometric structure. . . . 82

Sean Bohun et al.
9 Towards the unsupervised learning of novel RFI sources. 97

1 Content classification and keyword extraction for ici.radio-canada.ca

David Alfonso [a]
Shivendra Bhardwaj [a]
Ilan Elbaz [a]
Abbass Ghaddar [a]
Fabrizio Gotti [a]
Philippe Langlais [a]
Guillaume Le Berre [a]
Vincent Letard [a]
Peng Lu [a]
Olivier Salaün [a]
Jason Jiechen Wu [a]
Laura Elisa Salas [b]
Vincent Barnabé-Lortie [c]

[a] *RALI, Université de Montréal, Montréal (Québec), Canada*

[b] *DIRO, Université de Montréal, Montréal (Québec), Canada*

[c] *Médias numériques, Radio-Canada, Montréal (Québec), Canada*

October 2020
Les Cahiers du GERAD

1.1 Introduction

Radio-Canada publishes between 450 and 600 news articles in French per day on the website *ici.radio-canada.ca* (https://ici.radio-canada.ca/). To facilitate their publication on Radio-Canada's platforms, these articles are annotated by the authors with one of 26 themes (e.g. *sports*, *politics*) and optionally annotated using 464 subthemes (e.g. *cricket*, *hockey*, *provincial politics*, *federal politics*). The themes and subthemes help both the human reader and the search engine crawlers find information in the correct domain of interest, browse multiple news connected by themes, and thematically limit the field of research of a specific article.

Tagging articles with themes and subthemes can be a delicate and time-consuming task. The journalists and content editors who are in charge of choosing the theme might not always agree on how to classify a particular article. Moreover, depending on the date, context, and personal preference, one slightly ambiguous article can be classified in a non-evident theme. Subthemes are very numerous, and sifting through them can become tiresome. Consequently Radio-Canada wishes to improve the consistency of their classification and reduce the amount of effort for journalists and content editors by proposing automatically detected themes and subthemes.

In addition Radio-Canada wishes to enrich the content description of each article with keywords. Because the idea of having keywords was not introduced in the journalistic domain until very recently, in their current state, *ici.radio-canada.ca* articles do not include them. We were therefore also tasked with the automatic extraction of keywords from the article content. The ideal solution would allow not only to offer keyword suggestions to journalists and content editors but also to label automatically the articles made available to us.

For this task Radio-Canada provided us with 901,156 news articles in French taken from a database export for the content published over the last few years.

1.2 Pre-workshop data preparation

For the sake of efficiency, and to avoid consuming too much of workshop time distributing, reading, and formatting the data provided, some preparation was carried out before the workshop. Vincent Barnabé-Lortie and Fabrizio Gotti worked extensively on the data during the week prior to the workshop. The latter facilitated the acquisition of the data, saved it in a convenient format on a RALI server, and developed a dedicated Application Programming Interface (API) for efficient access to the data. He also acquired some potentially useful auxiliary data.

1.3 Data analysis

Before we could translate the industry problems into implementable tasks and sub-tasks, we needed to look at the data and understand what were the tools and raw material at our disposal.

With the help of Vincent Barnabé-Lortie, we deduced the following regarding both the structure of the data/meta-data and the journalists' procedures when submitting an article:

- Each article mainly consists of a title, a summary, a lead paragraph, the body (all the paragraphs), the theme and, optionally, the subtheme.
- Some articles may be empty for some of the sections mentioned above: this mainly happens in the case of news tickers, which do not have the traditional structure of an article per se.
- Some of the themes and subthemes in the taxonomy are deprecated and no longer used, even though they do appear in the data corpus.
- The journalists and content editors are required to specify one theme (and only one) for the article among the existing options.
- One of the themes in the taxonomy is called *Aucun thème sélectionné* ("No theme selected") and it seems to be used for articles written before the theme/subtheme taxonomy was fully established.

- It is not mandatory for the journalists and content editors to specify a subtheme, but there is virtually no limit on the number of subthemes an article may contain.
- There is no procedure that allows the journalist or content editor to propose a list of keywords for each article.
- One of the tools provided by Radio-Canada was a dictionary linking subthemes to themes, but we observed that this theme/subtheme link was not always respected (as further shown in Section 1.4).

1.4 Some statistics

Before continuing with the description of the tasks, we present some statistics on the data, which served as a guide in the resolution of the problems at hand. The observations derived from Figures 1.1, 1.2, 1.3, 1.4, and 1.5 led us to formulate the methods required to clean the data.

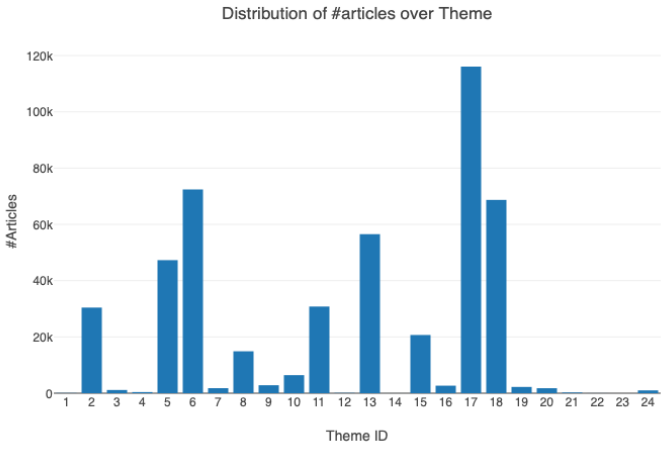


Figure 1.1: Distribution of the number of articles over the themes, showing some themes are not active in the whole data set.

As shown in Figure 1.1 some themes are not active or have a small frequency. This is also the case for subthemes. This means that the distribution is unequal and possibly biased towards certain themes, as we describe in the following sections.

In Figure 1.3 we observe that the subthemes labelling is very imbalanced. The top three most frequent subthemes are *Hockey*, *Politique provinciale*, and *Éducation*.

It is worth noting that some subthemes are not linked to one specific theme but to several themes (e.g. the subtheme *Mental health* is often linked to the themes *Health*, *Society*, *Miscellaneous news*, etc.). This is further analyzed in Figures 1.4 and 1.5.

1.5 Data cleaning and splitting

As a group we agreed on the need to clean the data and use the same subsets to train our models and evaluate ourselves if we wanted our results to be comparable.

As we mentioned in Section 1.3, providing a subtheme is not mandatory for publishing an article on the site *ici.radio-canada.ca*. This means that a lot of the articles have a theme but no subtheme. In order to work on an equal footing when using themes and subthemes classifiers, we decided to remove all articles that did not contain both a theme and (at least) one subtheme.

There is one theme that corresponds to “No theme selected.” Since this null-theme matches no specific domain, we chose to remove all articles labeled with this theme from our data set.

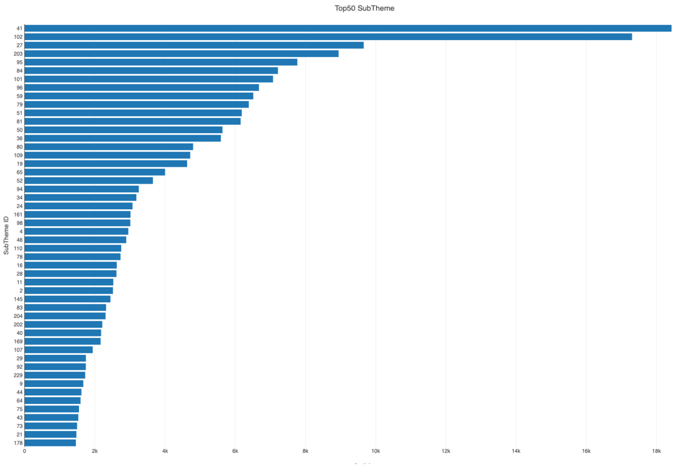


Figure 1.2: Excerpt of the distribution of the 50 most frequent subthemes over the number of articles, showing the most common subthemes in the data set.

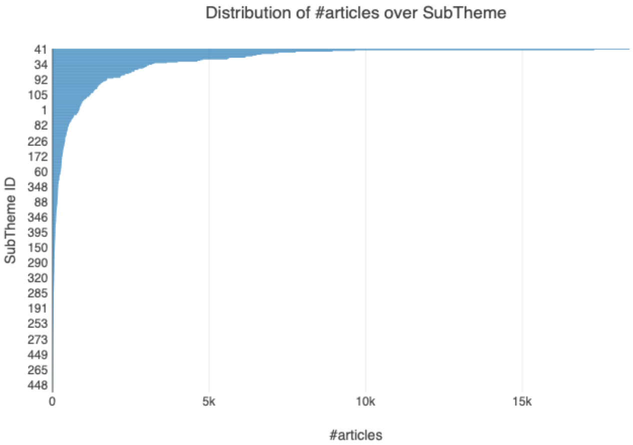
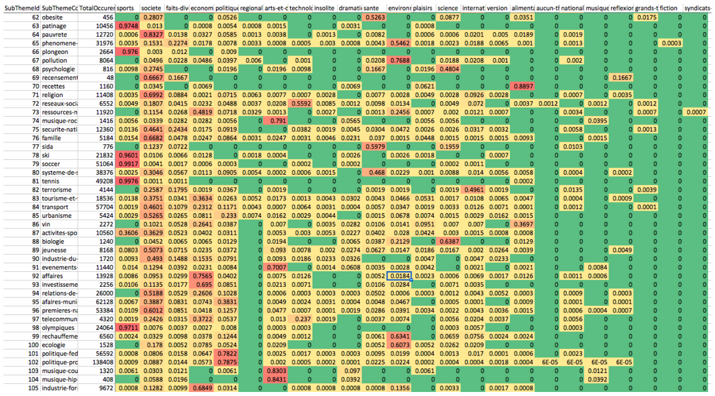


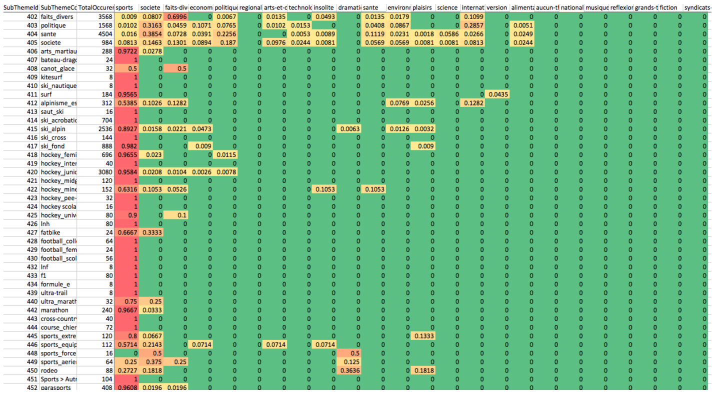
Figure 1.3: Sorted distribution of the subthemes over the number of articles, showing that the distribution of subthemes is also imbalanced.



SubThemeId	SubThemeCo	TotalOccurer	sports	societe	faits-div	econom	politique	regional	arts-et-c	technolo	insolite	dramati	sante	environ	plaisirs	science	internat	version	alimenta	aucun-th	national	musique	reflexion	grands-t	fiction	syndicats-
62	obesite	456	0	0.2807	0	0	0.0526	0	0	0	0	0	0.5263	0	0	0.0877	0	0	0.0351	0	0	0	0	0.0175	0	0
63	patinage	10456	0.9748	0.013	0	0.0008	0.0054	0.0008	0.0015	0	0	0.0031	0	0	0.0008	0	0	0	0	0	0	0	0	0	0	0
64	pauvrete	12720	0.0006	0.8327	0.0138	0.0327	0.0585	0.0013	0.0038	0	0.0013	0	0.0082	0	0.0006	0.0006	0.0201	0.005	0.0189	0	0.0019	0	0	0	0	0
65	phenomene-	31976	0.0035	0.1531	0.2274	0.0178	0.0078	0.0033	0.0008	0.0005	0.003	0.0008	0.0043	0.5462	0.0018	0.0023	0.0188	0.0065	0.001	0	0.0013	0	0	0	0.0003	0
66	plongeon	2664	0.976	0.003	0.012	0	0.009	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
67	pollution	8064	0	0.0496	0.0228	0.0486	0.0397	0.006	0	0.001	0	0	0.0208	0.7688	0	0.0188	0.0208	0.001	0	0	0.002	0	0	0	0	0
68	psychologie	816	0.0098	0.2745	0	0	0.0196	0	0.0196	0.0098	0	0	0.1667	0	0.0196	0.4804	0	0	0	0	0	0	0	0	0	0
69	recensement	48	0	0.6667	0.1667	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.1667	0	0	0
70	recettes	1160	0	0.0345	0	0.0069	0	0	0	0	0	0.0069	0	0	0.0621	0	0	0	0.8897	0	0	0	0	0	0	0
71	religion	11408	0.0035	0.6992	0.0884	0.0021	0.0715	0.0063	0.0077	0.0007	0.0028	0	0.0077	0.0028	0.0049	0.0028	0.0926	0.0028	0	0	0.0007	0	0.0035	0	0	0
72	reseaux-soci	6552	0.0049	0.1807	0.0415	0.0232	0.0488	0.0037	0.0208	0.5592	0.0085	0.0012	0.0098	0.0134	0	0.0049	0.072	0	0.0037	0.0012	0	0.0012	0	0.0012	0	0
73	ressources-n	11920	0	0.1154	0.0268	0.4819	0.0718	0.0329	0.0013	0	0.0027	0	0.0013	0.2456	0.0007	0.002	0.0121	0.0007	0	0	0.0034	0	0	0.0007	0	0.0007
74	musique-roc	1416	0.0056	0.0339	0.0282	0.0282	0.0056	0	0.791	0	0	0.0565	0	0	0.0056	0.0056	0	0	0	0	0	0.0395	0	0	0	0
75	securite-nati	12360	0.0136	0.4641	0.2434	0.0175	0.0919	0	0	0.0382	0.0019	0.0045	0.0304	0.0472	0.0026	0.0026	0.0317	0.0032	0	0	0.0058	0	0	0.0013	0	0
76	famille	5184	0.0154	0.6682	0.0478	0.0247	0.0864	0.0031	0.0247	0.0031	0.0046	0.0231	0.037	0.0015	0.0448	0.0015	0.0015	0.0015	0.0093	0	0	0.0015	0	0	0	0
77	sida	776	0	0.1237	0.0722	0	0	0	0	0	0	0	0.5979	0	0	0.1959	0	0	0	0	0.0103	0	0	0	0	0
78	ski	21832	0.9601	0.0106	0.0066	0.0128	0	0.0018	0.0004	0	0	0.0026	0	0.0026	0.0018	0	0	0.0007	0	0	0	0	0	0	0	0
79	soccer	51064	0.9917	0.0041	0.0017	0.0006	0.0003	0	0	0.0002	0	0.0002	0	0	0	0.0002	0.0011	0	0	0	0	0	0	0	0	0
80	systeme-de-s	38376	0.0025	0.3046	0.0567	0.0113	0.0905	0.0054	0.0002	0.0006	0.0015	0	0.468	0.0229	0.001	0.0088	0.014	0.0056	0.0058	0	0.0004	0	0.0002	0	0	0
81	tennis	49208	0.9976	0.0011	0.0011	0	0	0	0	0	0	0	0	0	0	0	0	0	0.0002	0	0	0	0	0	0	0
82	terrorisme	4144	0	0.2587	0.1795	0.0019	0.0367	0	0.0019	0	0	0	0.0019	0.0019	0	0.0019	0.4961	0.0019	0	0	0.0135	0	0	0.0039	0	0
83	tourisme-et-	18536	0.0138	0.3751	0.0341	0.3634	0.0263	0.0052	0.0173	0.0013	0.0043	0.0302	0.0043	0.0466	0.0531	0.0017	0.0108	0.0065	0.0047	0	0.0004	0	0.0009	0	0	0
84	transport	57704	0.0019	0.4601	0.1079	0.2312	0.1171	0.0043	0.0007	0.0064	0.0031	0.0004	0.0057	0.0347	0.0019	0.0033	0.0126	0.0071	0.0001	0	0.0012	0	0	0.0001	0	0
85	urbanisme	5424	0.0029	0.5265	0.0265	0.0811	0.233	0.0074	0.0162	0.0029	0.0044	0	0.0015	0.0678	0.0074	0.0015	0.0029	0.0162	0.0015	0	0	0	0	0	0	0
86	vin	2272	0	0.1021	0.0528	0.2641	0.0387	0	0.007	0	0.0035	0.0282	0.0106	0.0141	0.0951	0.007	0.007	0	0.3697	0	0	0	0	0	0	0
87	activites-spo	10560	0.3606	0.3629	0.0523	0.0402	0.0311	0	0.0053	0.003	0.0053	0.0227	0.0402	0.028	0.0424	0.003	0.0015	0.0008	0.0008	0	0	0	0	0	0	0
88	biologie	1240	0	0.0452	0.0065	0.0065	0.0129	0	0.0194	0	0	0.0065	0.0387	0.2129	0	0.6387	0	0	0.0129	0	0	0	0	0	0	0
89	jeunesse	8168	0.0803	0.5073	0.0715	0.0235	0.0372	0	0.093	0.0078	0.002	0.0274	0.0627	0.0147	0.0186	0.0167	0.002	0.0264	0.0039	0	0	0	0.0049	0	0	0
90	industrie-du-	1720	0.0093	0.493	0.1488	0.1535	0.0791	0	0.0093	0.0186	0.0233	0.0326	0	0	0.0047	0	0.0047	0.0233	0	0	0	0	0	0	0	0
91	evenements-	11440	0.014	0.1294	0.0392	0.0231	0.0084	0	0.7007	0	0.0014	0.0608	0.0035	0.0028	0.0042	0	0.0021	0	0.0021	0	0	0.0084	0	0	0	0
92	affaires	13928	0.0086	0.0953	0.0299	0.7565	0.0402	0	0.0075	0.0126	0	0	0.0052	0.0184	0.0023	0.0006	0.0069	0.0017	0.0126	0	0.0011	0.0006	0	0	0	0
93	investisseme	2256	0.0106	0.1135	0.0177	0.695	0.0851	0	0.0213	0.0071	0	0	0.0106	0.0284	0	0.0071	0.0035	0	0	0	0	0	0	0	0	0
94	relations-de-	26000	0	0.5188	0.0529	0.2606	0.1028	0	0.0006	0.0003	0.0003	0.0003	0.0502	0.0006	0.0003	0.0012	0.0043	0.0052	0.0003	0	0.0009	0	0.0003	0	0	0
95	afaires-muni	62128	0.0067	0.3887	0.0831	0.0743	0.3831	0	0.0049	0.0024	0.0031	0.0004	0.0048	0.0467	0	0.0005	0.0001	0	0.0009	0	0.0001	0	0.0001	0	0	0
96	premieres-na	53384	0.0109	0.6012	0.0851	0.0418	0.1257	0	0.0477	0.0007	0.0001	0.0019	0.0286	0.0391	0.0034	0.0022	0.0043	0.0036	0.0015	0	0.0006	0.0004	0.0007	0	0	0
97	telecommun	4320	0.0019	0.2426	0.0315	0.3722	0.0537	0	0.013	0.237	0.0019	0	0.0037	0.0074	0	0.0056	0.0204	0.0037	0	0	0.0056	0	0	0	0	0
98	olympiques	24064	0.9711	0.0076	0.0037	0.0027	0.008	0	0.0003	0.0003	0	0	0	0	0	0.0003	0.0057	0	0	0	0.0003	0	0	0	0	0
99	rechauffeme	6560	0.0024	0.0329	0.0098	0.0378	0.1244	0	0.0049	0.0012	0	0	0.0061	0.6341	0	0.0659	0.0756	0.0024	0.0024	0	0	0	0	0	0	0
100	ecologie	1528	0	0.178	0.0052	0.0785	0.0524	0	0.0209	0	0	0	0.0052	0.6073	0.0052	0.0262	0.0209	0	0	0	0	0	0	0	0	0
101	politique-fed	56592	0.0008	0.0806	0.0158	0.0647	0.7822	0	0.0025	0.0017	0.0003	0.0003	0.0095	0.0199	0.0004	0.0013	0.017	0.0001	0.0006	0	0.0023	0	0	0	0	0
102	politique-pro	138408	0.0009	0.0887	0.0144	0.0573	0.7875	0	0.002	0.0005	0.0002	0.0001	0.0225	0.0224	0.0002	0.0004	0.0004	0.0018	0.0004	6E-05	6E-05	6E-05	6E-05	0	0	0
103	musique-cou	1320	0.0061	0.0303	0.0121	0	0.0061	0	0.8303	0	0	0.097	0	0	0.0061	0	0	0	0	0	0	0.0121	0	0	0	0
104	musique-hip-	408	0	0.0588	0.0196	0	0	0	0.8431	0	0	0.0392	0	0	0	0	0	0	0	0	0	0.0392	0	0	0	0
105	industrie-for	9672	0.0008	0.1282	0.0099	0.6849	0.0314	0	0	0.0008	0.0008	0.0008	0.0008	0.1356	0	0.0033	0	0.0017	0.0008	0	0	0	0	0	0	0

Figure 1.4: Excerpt of the heat-map of the themes (horizontal) and subthemes (vertical) correspondence, showing subthemes spread over numerous themes. This indicates that some subthemes are not limited to specific themes.

After cleaning the total number of articles fell from around 900,000 to approximately 240,000 (25%). Then the remaining data was split into three parts: the training set, the validation set, and the test set.



SubThemeId	SubThemeCo	TotalOccurer	sports	societe	faits-div	econom	politique	regional	arts-et-c	technolo	insolite	dramati	sante	environ	plaisirs	science	internat	version	alimenta	aucun-th	national	musique	reflexior	grands-t	fiction	syndicats-
402	faits_divers	3568	0.009	0.0807	0.6996	0	0.0067	0	0.0135	0	0.0493	0	0.0135	0.0179	0	0	0.1099	0	0	0	0	0	0	0	0	0
403	politique	1568	0.0102	0.3163	0.0459	0.1071	0.0765	0	0.0102	0.0153	0	0	0.0408	0.0867	0	0	0.2857	0	0.0051	0	0	0	0	0	0	0
404	sante	4504	0.016	0.3854	0.0728	0.0391	0.2256	0	0	0.0053	0.0089	0	0.1119	0.0231	0.0018	0.0586	0.0266	0	0.0249	0	0	0	0	0	0	0
405	societe	984	0.0813	0.1463	0.1301	0.0894	0.187	0	0.0976	0.0244	0.0081	0	0.0569	0.0569	0.0081	0.0081	0.0813	0	0.0244	0	0	0	0	0	0	0
406	arts_martiau	288	0.9722	0.0278	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
407	bateau-drag	24	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
408	canot_glace	32	0.5	0	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
409	kitesurf	8	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
410	ski_nautique	8	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
411	surf	184	0.9565	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.0435	0	0	0	0	0	0	0	0
412	alpinisme_es	312	0.5385	0.1026	0.1282	0	0	0	0	0	0	0	0	0.0769	0.0256	0	0.1282	0	0	0	0	0	0	0	0	0
413	saut_ski	16	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
414	ski_acrobatic	704	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
415	ski_alpin	2536	0.8927	0.0158	0.0221	0.0473	0	0	0	0	0	0.0063	0	0.0126	0.0032	0	0	0	0	0	0	0	0	0	0	0
416	ski_cross	144	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
417	ski_fond	888	0.982	0	0	0.009	0	0	0	0	0	0	0	0	0.009	0	0	0	0	0	0	0	0	0	0	0
418	hockey_femi	696	0.9655	0.023	0	0	0.0115	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
419	hockey_inter	40	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
420	hockey_juni	3080	0.9584	0.0208	0.0104	0.0026	0.0078	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
421	hockey_midg	120	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
422	hockey_mine	152	0.6316	0.1053	0.0526	0	0	0	0	0	0.1053	0	0.1053	0	0	0	0	0	0	0	0	0	0	0	0	0
423	hockey_pee-	32	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
424	hockey scola	16	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
425	hockey_univ	80	0.9	0	0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
426	lnh	80	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
427	fatbike	24	0.6667	0.3333	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
428	football_coll	64	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
429	football_fem	24	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
430	football_scol	56	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
432	lnf	8	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
433	f1	80	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
434	formule_e	8	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
439	ultra-trail	8	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
440	ultra_marath	32	0.75	0.25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
442	marathon	240	0.9667	0.0333	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
443	cross-countr	40	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
444	course_chier	72	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
445	sports_extre	120	0.8	0.0667	0	0	0	0	0	0	0	0	0	0	0.1333	0	0	0	0	0	0	0	0	0	0	0
446	sports_equip	112	0.5714	0.2143	0	0.0714	0	0	0.0714	0	0.0714	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
448	sports_force	16	0	0.5	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0
449	sports_aerie	64	0.25	0.375	0.25	0	0	0	0	0	0	0.125	0	0	0	0	0	0	0	0	0	0	0	0	0	0
450	rodeo	88	0.2727	0.1818	0	0	0	0	0	0	0	0.3636	0	0	0.1818	0	0	0	0	0	0	0	0	0	0	0
451	Sports > Autr	104	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
452	parasports	408	0.9608	0.0196	0.0196	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 1.5: Excerpt of the heat-map of the themes (horizontal) and subthemes (vertical) correspondence, showing subthemes very focused on one or very few themes. This indicates that some subthemes are rather exclusive to a specific theme.

After having randomized the whole data set, of the total of 240,000 articles 80% were assigned to the training set, 10% to the validation set, and another 10% to the test set. This was carried out once and distributed to the whole team so we could train, validate, and test each model on the same subsets.

We also chose a specific output format of the various classifiers we tested and an evaluation protocol for which an algorithm was implemented (in order to ensure evaluation uniformity across all the sub-teams). Despite having spent one week to prepare the data before the workshop, the team spent more than a day to choose a way to clean the data and to do the actual cleaning.

1.6 Theme and subtheme classifiers

The first task consisted in training classifiers in a supervised way in order to predict the theme or the subthemes of an article, based on its text content.

Due to time constraints, we tested two well-established feature-based approaches, namely Logistic Regression and Support Vector Machines (Cortes et Al. [2]), as well as two approaches based on deep learning, namely BERT (Devlin et al. [3]) and fastText (Joulin et al. [4]). It is important to stress that many of those algorithms are controlled by hyper-parameters that we did not have time to investigate in depth.

All resulting scores are shown in Table 1.1.

Table 1.1: Classification task scores.

Sub-task	Classifier	Precision @1	Recall @1	F1 @1
Theme identification	BERT	75.4%	75.4%	75.4%
Theme identification	fastText	77.6%	77.6%	77.6%
Theme identification	Logistic regression	77.8%	77.8%	77.8%
Theme identification	**SVM**	**78.7%**	**78.7%**	**78.7%**
subtheme identification	BERT	7.1%	5.8%	6.4%
subtheme identification	**fastText**	**71.1%**	**58.1%**	**63.9%**
subtheme identification	Logistic regression	n/a	n/a	n/a
subtheme identification	SVM	n/a	n/a	n/a

1.6.1 Logistic regression

We first experimented with a classifier built on logistic regression (logit). To do so, we converted each article's text to a bag-of-words representation, in this case a matrix of TF-IDF features. More precisely the title, lead, and body of each article are concatenated and tokenized; then the frequency of each word is recorded.

A vector representation of this bag-of-words is produced, where each dimension corresponds to a given word. Each article is therefore represented by a vector of dimension V, the size of the vocabulary minus stop words, which were ignored. The value of each coefficient is computed with the standard TF-IDF score. Very roughly this means that frequent words in a document get rewarded by this score if they are not too frequent in the other articles. Training such a model amounts to learning to weigh each dimension (word) of the associated vector space representation so as to minimize classification error. The advantage of such a technique is that we can observe the learned weights and see what the model thinks about the importance of a specific word for a given theme (or subtheme). We used Scikit-Learn (nearly in its default setting) for training the model. A modest number of hyper-parameters were tried but they yielded very closely grouped performance results, ultimately.

1.6.2 SVM

Support Vector Machines have often been reported as robust classifiers. Therefore we tested one such approach. The feature representation was very similar to the one given to the Logistic Regression method. Here we attempted to boost the weight of certain excerpts of the article (notably the summary and title), reasoning that the words in these sections are more informative, but to no avail. The TF-IDF vectorization scheme proved to be of little help and the raw counts yielded very acceptable results, reported here.

We used the implementation in Scikit-Learn and its `SGDClassifier` classifier with almost all of the default arguments except for the random state value (42), the maximum of iterations (limited to 16), and the tolerance (None). Notably the number of iterations had the greatest impact on performances.

This model yielded the best accuracy as a theme classifier. It is worth noting that both the Logistic Regression and SVM models can run on ordinary CPUs and do not require GPUs (graphical processing units).

1.6.3 BERT

BERT (Devlin et al. [3]), an acronym meaning Bidirectional Encoder Representations from Transformers, leverages a powerful bidirectional Transformer (an attention model) in order to tackle a remarkable variety of NLP tasks with state-of-the-art results, including sentiment analysis and question answering.

BERT is a pre-trained model. It provides contextual embeddings of the words (or fragments of words) in a sentence and can ultimately be used to extract a vector representation of a sequence of tokens. For a particular task the model is fine-tuned in order to obtain a representation better tailored to the task, while its output is fed to a small additional layer that performs classification, for instance.

For the theme/subthemes classification task, we gave BERT the first 128 tokens of the article body and took the embedding of the first token (corresponding to the “Start” symbol) to be our sentence representation. This representation is then fed to two distinct linear layers followed by softmax layers (for themes and subthemes). We used a cross-entropy loss for training. For subthemes experiments were made using a multi-label objective but this method showed really weak results and we finally chose a cross-entropy loss using only the first subtheme as gold-standard label.

Training is carried out using Adam with a learning rate of 10^{-5}. For these experiments we used a pytorch implementation of BERT (pytorch-transformers) and the `bert-base-multilingual-cased` pre-trained model. The latter supports multiple languages (including French).

1.6.4 fastText

Another popular and recent neural network model we tested is fastText, from Facebook’s AI Research lab (Joulin et al. [4]). The fastText model allows vector word representations based on the sum of their characters’ n-grams; it has been shown to be specially useful for text classification tasks where classes are imbalanced and when time is limited (which was the case within the context of the workshop). Its performances are often on a par with those of deep learning approaches but it can be trained much faster. Moreover one of the reasons fastText was created was to handle and leverage morphologically rich languages such as French.

We used the basic available model and chose, as hyper-parameters, a learning rate of 1.0, with an embedding vector dimension of 50, a word n-gram length of 2, a bucket value of 200,000, and we trained for 40 epochs. The classification task consists in representing each word by a vector of size 50 for a given article, then averaging them to produce a single vector representation of the article. This in turn is fed to a linear classifier with a hierarchical softmax.

This strategy allowed us to obtain a very good score for theme classification and the best score for subtheme classification.

Contrary to the Logistic Regression and SVM models, the BERT and fastText models must run on highly potent GPUs in order to obtain results relatively quickly. BERT was trained using four Nvidia RTX 2080 Ti and convergence was achieved after approximately 2 hours. The fastText model was trained using one Titan XP and convergence was achieved after approximately 4 hours.

1.6.5 Future work

An obvious conclusion drawn by the classifier sub-teams is that in order to gain a better understanding of the data, we need to make a better analysis of the labelling done by the journalists. Understanding their priorities when they label an article with a specific theme or subtheme is key to understanding how clean or uniform the data is.

For the theme classification task the best performing model was SVM. We expected it to outperform the Logistic Regression model but we were surprised by the lower performance of the deep learning approaches. Several reasons can explain this, among which the relatively small training set and the multilingual embeddings we used to seed the model.

There is yet room for improvement. Even with our best efforts there is only so much we can do in just one week of work with the data. Our highest F1 score for theme (resp. sub theme) classification is 78.7% (resp. 63.9%). We believe these scores can be improved by testing other models, better adjustment of the hyper-parameters, or even an automatic data cleaning for themes that are labelled ambiguously.

1.7 Keyword extraction

In order to allow an improved article access to the potential reader, the theme is not always enough. This is because themes and subthemes only represent a more or less general thematic domain while, very often, the reader has a greater interest in a specific subject. This is where keywords become useful.

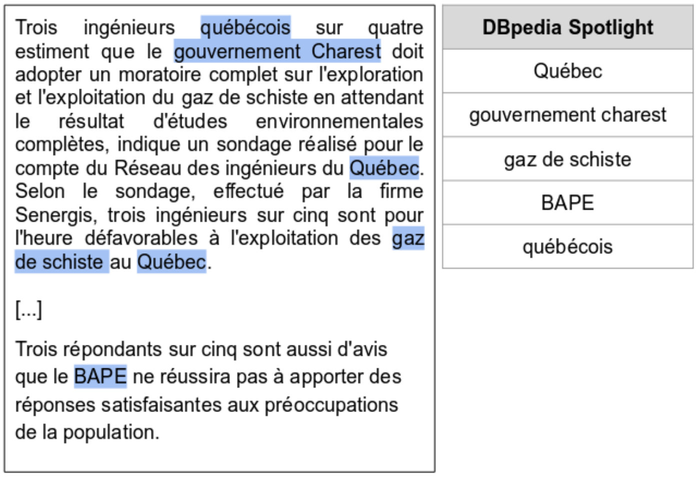
Since the database does not have any keyword annotations and we lack a large number of readers-annotators to extract or pinpoint keywords manually or label each one of the 900,000 articles with multiple keywords, we chose to design an automatic keyword extraction method. This method should analyze the content of the article in order to deduce what words (or groups of words) would be representative keywords.

The greatest difficulty we encountered when designing this method was its evaluation. Without a keyword-annotated corpus we were unable to produce any evaluative test other than the superficial human analysis of some very limited examples. We still spent some time deploying mainly two core technologies, which we describe below.

1.7.1 DBpedia Spotlight

DBpedia Spotlight (Mendes et al. [5]) is an open-source free tool that allows to connect text to existing entries of Wikipedia and DBpedia. Basically, by giving the body of the article to the tool, we are able to extract the words and groups of words that appear both in the article and as encyclopedic/ontological entries in those resources.

After running this tool we were able to extract 14,867,719 keywords from the 901,156 articles. This represents an average of 16.5 keywords per article. An excerpt of an article with the extracted keywords can be seen in Figure 1.6.



Trois ingénieurs québécois sur quatre estiment que le gouvernement Charest doit adopter un moratoire complet sur l'exploration et l'exploitation du gaz de schiste en attendant le résultat d'études environnementales complètes, indique un sondage réalisé pour le compte du Réseau des ingénieurs du Québec. Selon le sondage, effectué par la firme Senergis, trois ingénieurs sur cinq sont pour l'heure défavorables à l'exploitation des gaz de schiste au Québec.

[...]

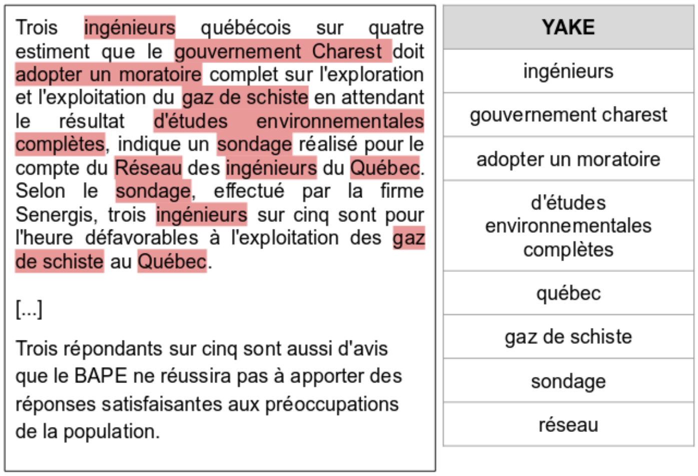
Trois répondants sur cinq sont aussi d'avis que le BAPE ne réussira pas à apporter des réponses satisfaisantes aux préoccupations de la population.

DBpedia Spotlight
Québec
gouvernement charest
gaz de schiste
BAPE
québécois

Figure 1.6: Excerpt of an article and keywords extracted by DBpedia Spotlight.

1.7.2 YAKE

Yet Another Keyword Extractor (YAKE) (Campos et al., [1]) is also a free open-source tool to extract keywords. The main difference is that YAKE is completely language independent and instead of mapping words to encyclopedic entries, is based on the word frequencies, their locations in sentences, named entities indicators, etc. This means that it extracts much more potential keywords but also much more noise as can be seen in Figure 1.7.



Trois ingénieurs québécois sur quatre estiment que le gouvernement Charest doit adopter un moratoire complet sur l'exploration et l'exploitation du gaz de schiste en attendant le résultat d'études environnementales complètes, indique un sondage réalisé pour le compte du Réseau des ingénieurs du Québec. Selon le sondage, effectué par la firme Senergis, trois ingénieurs sur cinq sont pour l'heure défavorables à l'exploitation des gaz de schiste au Québec.

[...]

Trois répondants sur cinq sont aussi d'avis que le BAPE ne réussira pas à apporter des réponses satisfaisantes aux préoccupations de la population.

YAKE
ingénieurs
gouvernement charest
adopter un moratoire
d'études environnementales complètes
québec
gaz de schiste
sondage
réseau

Figure 1.7: Excerpt of an article and keywords extracted by YAKE.

1.7.3 Future work

Concerning the keyword extraction task, we already have results but we lack the evaluation tools to test them. Had we had more time, we could have explored some ways of testing them.

One way would have been to run both these tools on keyword-labeled data and compare the output results with the human labels. Our guess is that this method would have had a very low precision score, because the algorithms are not able to detect which keywords are representative of the whole article as well as a human annotator can.

Another evaluation strategy we could have used is human evaluation. By detecting all the keywords from the keywords extraction tools and having them analyzed by a jury of annotators, we would have been able to determine with a great precision whether the method is up to human standards. Nevertheless this would have been extremely time-consuming and costly since we would have needed at the very least three human annotators working on a sample set big enough to be representative. Each annotator would have had to read every article and analyze the keywords that were extracted.

Another choice would have been to annotate our corpus with the keywords from each system and to see whether they improve the results from our theme/subtheme classification task.

1.8 Conclusion

We had two tasks for this workshop: classify articles by themes and subthemes based on their content and automatically extract keywords. For the first task we proposed four different systems rendering very close results: one based on Google's BERT model, one based on Facebook's fastText model, one based on the logistic regression approach, and one based on the SVM approach. The classification sub-tasks were evaluated by comparing the top prediction of the model to the human-annotated label. For the theme classification sub-task, the SVM returns the best result of all but by a 3% difference only. For the subtheme classification problem, we experimented with only two out of the four methods (fastText yielded the best result).

The theme/subtheme classifiers help the journalists and content editors and save them the time to search potential themes and subthemes to classify each new article. The keyword extractor allows one to add content metadata to the already existing article database. Both tasks could be further improved in many ways but this would require an even more profound analysis of the data annotation and an implementation of a keyword extraction evaluation.

Bibliography

[1] Campos, Ricardo and Mangaravite, Vítor and Pasquali, Arian and Jorge, Alípio Mário and Nunes, Célia and Jatowt, Adam. A text feature based automatic keyword extraction method for single documents. In European Conference on Information Retrieval, pages 684–691. Springer, 2018.

[2] Cortes, Corinna and Vapnik, Vladimir. Support-vector networks. In Machine learning, volume 20, number 3, pages 273–297. Springer, 1995.

[3] Devlin, Jacob and Chang, Ming-Wei and Lee, Kenton and Toutanova, Kristina. Bert: Pre-training of deep bidirectional transformers for language understanding. In arXiv preprint, arXiv:1810.04805, 2018.

[4] Joulin, Armand and Grave, Edouard and Bojanowski, Piotr and Mikolov, Tomas. Bag of tricks for efficient text classification. In arXiv preprint, arXiv:1607.01759, 2016.

[5] Mendes, Pablo N and Jakob, Max and García-Silva, Andrés and Bizer, Christian. DBpedia spotlight: shedding light on the web of documents. In Proceedings of the 7th international conference on semantic systems, pages 1–8. ACM, 2011.

2 Role-oriented legal information retrieval

Pan Du [a]
Yifan Nie [a]
Louis Lv [a]
Jian-Yun Nie [a]
François Mercier [b]
Simon Picard [b]

[a] *RALI, Université de Montréal, Montréal (Québec), Canada*

[b] *Autorité des Marchés Financiers, Montréal (Québec), Canada*

October 2020
Les Cahiers du GERAD

Abstract: *This report describes some attempts to alleviate the burden of end users in their search for relevant regulations and law clauses within the official documents hosted by the Autorité des Marchés Financiers (AMF). We implemented a role-oriented interactive search engine for legal document retrieval. This specialized search engine, different from a general search engine, addresses two specific problems: (i) the domain-specific term-mismatch problem, and (ii) the role-oriented scope search problem. To address these problems we first extracted from the legal documents a taxonomy providing associations between the concepts. This taxonomy is used to expand a user's queries. Scopes for various roles in the legal field were also extracted and the legal documents were segmented based on the roles. This allows a search to be more restricted: a query for a role will only retrieve a list of results that are relevant for the role. The search engine was built upon a popular general-purpose open-source search engine (Indri) and includes the proposed enhancements. This work tested the feasibility of some approaches to legal document retrieval for the AMF; it can lead to further developments.*

2.1 Introduction

The mandate of the Autorité des Marchés Financiers (AMF) is to develop and monitor the application of financial sector regulations for Québec and foreign companies offering financial products in Québec. A number of official documents such as the legislation, regulations, notices, and other legal documents for securities professionals[1] are made available to the public on the AMF web site. In general these regulations have some structure and include in particular:

- definitions of various entities (for instance the definition of "derivative");
- articles;
- regulations and guidance;
- references to other regulations or legislative pieces.

In order to abide by the regulations, companies in the financial sector are invited to understand the regulations so as to ensure that they are correctly applied to their specific cases. At the present time this understanding exercise is mostly "manual," whether one is looking for the relevant sections or ensuring that the company is complying with the law. In general a financial sector company includes a compliance department, which ensures that the company complies with all the regulations (either the AMF regulations or those of other relevant institutions).

In order to lighten the regulatory burden, several projects have been launched in other countries by organizations similar to the AMF, for example, organizations in the United States and the United Kingdom. The current project also aims at making less burdensome the search for relevant information within the laws, regulations, guidelines, and so on.

Legal texts usually contain mandated definitions, to ensure the terminology is unambiguous: this is crucial when searching for some information. The rules contained in rule books vary according to clients and/or financial products: thus the scope of each section varies according to the regulator. Market participants, however, are usually subject to several regulators and thus several rule books. When a client is trying to find the rules that apply to a specific case, he requires some legal knowledge and skills but he also needs to read through the whole rule book to locate the applicable rules. The client is not always a professional who knows well the terminology and the structure of rule books: if he does not have the necessary knowledge, his search for useful information may be difficult. Even if the client has enough knowledge about the terminology and the structure of the rule books, reading through the documents to locate the applicable sections can be burdensome.

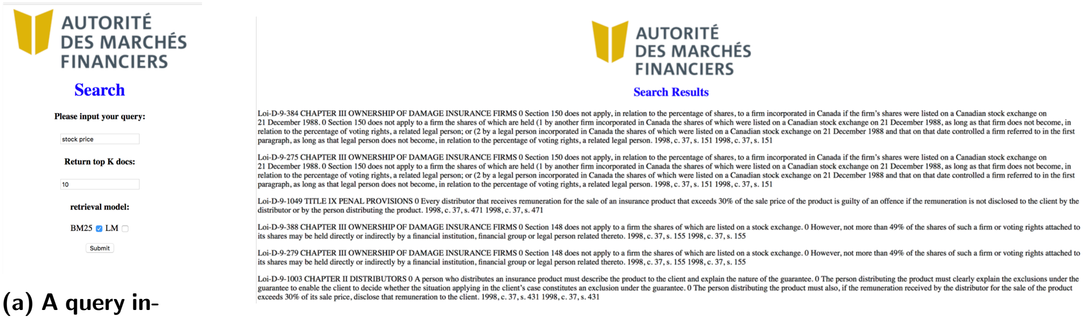
[1] https://lautorite.qc.ca/en/professionals/regulations-and-obligations/securities/

2.2 The problems

There are two major problems to be solved: the first one is related to term mismatch and the second one to role-oriented search. The terminology used in legal documents, i.e., the taxonomy, could be different from that used by a client of AMF (user). This is a key problem in accessing information that applies to a particular cases. Without a proper taxonomy it is difficult to extract useful information from legal documents: for example the client may use the phrase "damage evaluator" to get at the concept of "claims adjuster". The second problem (role-oriented text scope) arises because not all the sections of the legal materials are relevant to every client and/or financial product. Finding all sections containing a certain concept is time-consuming for the client. A further difficulty is that similar yet different informations may mislead the client in some cases. It is crucial to identify the sections that are relevant to the role the client is playing.

To address the above problems, we propose to design a role-oriented information retrieval framework enabling both professionals and non-professionals to search for legal information.

In a general search engine, as shown in Figure 2.1, the user typically inputs a query to describe the information that he (or she) needs and the search engine returns a list of results that may meet the need. When processing the query, the search engine does not take into account the scope or the role of the user: every user will get the same results. It may also modify the initial query in order to get "better" results for the user. These results, however, can be "better" only if the modification is done in an appropriate manner by considering domain-specific information, which is not guaranteed if one uses a general search engine. In summary one faces two key challenges when using such an engine.



Please input your query: stock price

Return top K docs: 10

retrieval model: BM25 ☑ LM □

(a) A query input.

Loi-D-9-384 CHAPTER III OWNERSHIP OF DAMAGE INSURANCE FIRMS 0 Section 150 does not apply, in relation to the percentage of shares, to a firm incorporated in Canada if the firm's shares were listed on a Canadian stock exchange on 21 December 1988. 0 Section 150 does not apply to a firm the shares of which are held (1 by another firm incorporated in Canada the shares of which were listed on a Canadian stock exchange on 21 December 1988, as long as that firm does not become, in relation to the percentage of voting rights, a related legal person; or (2 by a legal person incorporated in Canada the shares of which were listed on a Canadian stock exchange on 21 December 1988 and that on that date controlled a firm referred to in the first paragraph, as long as that legal person does not become, in relation to the percentage of voting rights, a related legal person. 1998, c. 37, s. 151 1998, c. 37, s. 151

Loi-D-9-275 CHAPTER III OWNERSHIP OF DAMAGE INSURANCE FIRMS 0 Section 150 does not apply, in relation to the percentage of shares, to a firm incorporated in Canada if the firm's shares were listed on a Canadian stock exchange on 21 December 1988. 0 Section 150 does not apply to a firm the shares of which are held (1 by another firm incorporated in Canada the shares of which were listed on a Canadian stock exchange on 21 December 1988, as long as that firm does not become, in relation to the percentage of voting rights, a related legal person; or (2 by a legal person incorporated in Canada the shares of which were listed on a Canadian stock exchange on 21 December 1988 and that on that date controlled a firm referred to in the first paragraph, as long as that legal person does not become, in relation to the percentage of voting rights, a related legal person. 1998, c. 37, s. 151 1998, c. 37, s. 151

Loi-D-9-1049 TITLE IX PENAL PROVISIONS 0 Every distributor that receives remuneration for the sale of an insurance product that exceeds 30% of the sale price of the product is guilty of an offence if the remuneration is not disclosed to the client by the distributor or by the person distributing the product. 1998, c. 37, s. 471 1998, c. 37, s. 471

Loi-D-9-388 CHAPTER III OWNERSHIP OF DAMAGE INSURANCE FIRMS 0 Section 148 does not apply to a firm the shares of which are listed on a stock exchange. 0 However, not more than 49% of the shares of such a firm or voting rights attached to its shares may be held directly or indirectly by a financial institution, financial group or legal person related thereto. 1998, c. 37, s. 155 1998, c. 37, s. 155

Loi-D-9-279 CHAPTER III OWNERSHIP OF DAMAGE INSURANCE FIRMS 0 Section 148 does not apply to a firm the shares of which are listed on a stock exchange. 0 However, not more than 49% of the shares of such a firm or voting rights attached to its shares may be held directly or indirectly by a financial institution, financial group or legal person related thereto. 1998, c. 37, s. 155 1998, c. 37, s. 155

Loi-D-9-1003 CHAPTER II DISTRIBUTORS 0 A person who distributes an insurance product must describe the product to the client and explain the nature of the guarantee. 0 The person distributing the product must clearly explain the exclusions under the guarantee to enable the client to decide whether the situation applying in the client's case constitutes an exclusion under the guarantee. 0 The person distributing the product must also, if the remuneration received by the distributor for the sale of the product exceeds 30% of its sale price, disclose that remuneration to the client. 1998, c. 37, s. 431 1998, c. 37, s. 431

(b) A example of results for the query input.

Figure 2.1: An example of the workflow of a general search engine.

- **Term Mismatch** When dealing with legal information retrieval, the legal concepts are important. For a client without adequate legal knowledge, the query input may not correspond to the concepts in the legal documents, leading to the so-called term mismatch problem. The results returned by a general search engine, in this case, will fail to meet the client's information need. A possible solution to this problem is to do a query expansion, which expands the original query according to the relationships between terms.
- **Role Independence** Another problem with a general search engine is that it neglects the role of the user. Certain regulations pertain to certain client's roles only. For example regulations for a damage insurance broker and a financial planner may be different. To return results that are acceptable to a client with a specific role, one can conduct a search within a scope: this involves scope segmentation and scope matching.

In the following we describe our attempt to address those two challenges.

2.3 Resources

The resources that the AMF offers are four types of laws and regulations (in the form of textual data), as shown in Table 2.1. They are, respectively: Securities, Insurance and deposit institutions, Distribution of financial products and services, and Derivatives. In this work (and for now) we focus on the English documents of Distribution of financial products and services. The proposed solution can be easily extended to the other three data sets, and to the French documents.

Table 2.1: Available resources.

name	Characters	Types
Securities	1 law, 59 regulations, 142 notices, Size: 536 MB, Format: Word, Languages: French and English	Loi, Règlementation
Insurance and deposit institutions	7 laws, 11 regulations, Size: 33 MB, Format: Html, PDF, Languages: French and English	Loi, Règlementation
Distribution of financial products and services	1 law, 20 regulations, 14 directives, Size: 35MB, Format: Html, PDF, Languages: French and English	Loi, Règlementation, Directive
Derivatives	1 law, 7 regulations, Size: 4 MB, Format: Html, Word, Languages: French and English	Loi, Règlementation

An instance of law is illustrated in Figure 2.2, where we display the first segment of a document containing some definitions: a representative, an insurance representative, and so on. These definitions are useful for extracting a taxonomy and implementing the framework of a legal document retrieval system.

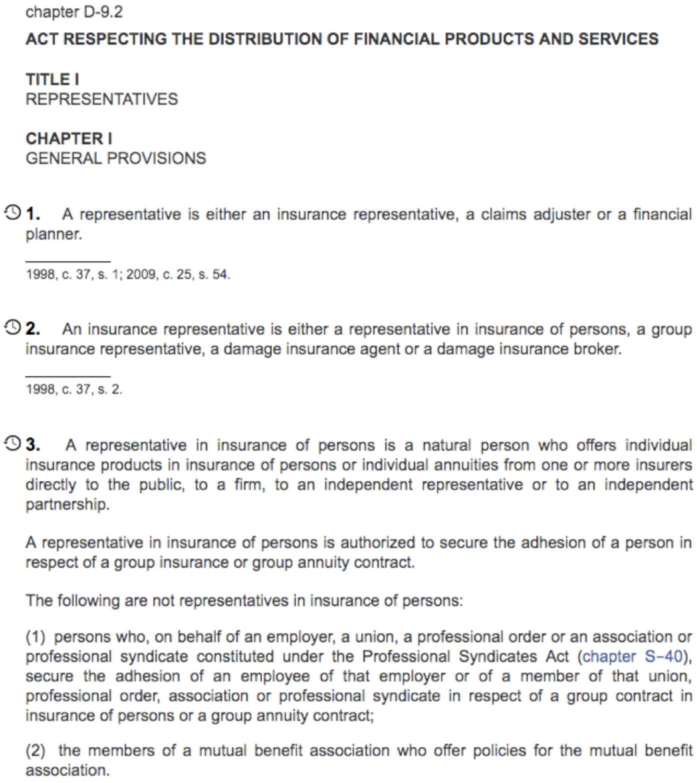
2.4 The approach we have chosen

Given the challenges and the resources, the proposed legal information retrieval framework consists mainly of three components: a taxonomy extractor, a scope identifier, and a search engine, as shown in Figure 2.3. The search engine includes three components: query expansion, role clarification, and relevance modelling.

Before the search engine can provide results for a given query, several steps of resource processing need to be carried out. The first step is the establishment of a taxonomy, which is essentially a set of concepts associated with one another. It needs to be constructed for the expansion of potential queries. The second step is the segmentation of scopes: the legal documents will be segmented according to the interests of different roles in this domain, so that only role-specific information will be provided when a search is launched.

2.4.1 Taxonomy extraction

A taxonomy is composed of key concepts as defined in the rule books. With such a concept set, query terms submitted to the legal document search engine can be expanded with terms from the taxonomy, so as to meet the information need more accurately.



chapter D-9.2

ACT RESPECTING THE DISTRIBUTION OF FINANCIAL PRODUCTS AND SERVICES

TITLE I
REPRESENTATIVES

CHAPTER I
GENERAL PROVISIONS

1. A representative is either an insurance representative, a claims adjuster or a financial planner.

1998, c. 37, s. 1; 2009, c. 25, s. 54.

2. An insurance representative is either a representative in insurance of persons, a group insurance representative, a damage insurance agent or a damage insurance broker.

1998, c. 37, s. 2.

3. A representative in insurance of persons is a natural person who offers individual insurance products in insurance of persons or individual annuities from one or more insurers directly to the public, to a firm, to an independent representative or to an independent partnership.

A representative in insurance of persons is authorized to secure the adhesion of a person in respect of a group insurance or group annuity contract.

The following are not representatives in insurance of persons:

(1) persons who, on behalf of an employer, a union, a professional order or an association or professional syndicate constituted under the Professional Syndicates Act (chapter S-40), secure the adhesion of an employee of that employer or of a member of that union, professional order, association or professional syndicate in respect of a group contract in insurance of persons or a group annuity contract;

(2) the members of a mutual benefit association who offer policies for the mutual benefit association.

Figure 2.2: A sample of rules in a law.

Candidate concept extraction

In this work, we adopt a rule-based approach to extracting the taxonomy. Rules based on language patterns for extracting candidate concepts can be divided into two major types:

- The **IS-A** Relation: for example, “is a,” “is an,” “is another,” “are,” etc. are all language patterns indicating a predication of two or more concepts connected by an “is-a” relation in the legal documents.
- The **PART-OF** Relation: patterns containing “include,” “contain,” “is composed of,” “is part of,” and so on, are used for identifying several candidate concepts in the context.

For example, as shown in Figure 2.4, a “damage insurance broker” is a “natural person,” where “damage insurance broker” and “natural person” are both candidate concepts for constructing the taxonomy. In Figure 2.5, a “discipline committee” is composed of “advocates” and “representatives,” where “discipline committee,” “advocates,” and “representatives” are all extracted as candidate concepts into the taxonomy.

Candidate concept filtering

The candidate concepts are then filtered according to certain criteria, such as term frequency (TF), document frequency (DF), TF-IDF (where IDF stands for “inverse data frequency”), and so on. Some borderline concepts are also labelled manually to ensure the precision of the concepts in the final taxonomy.

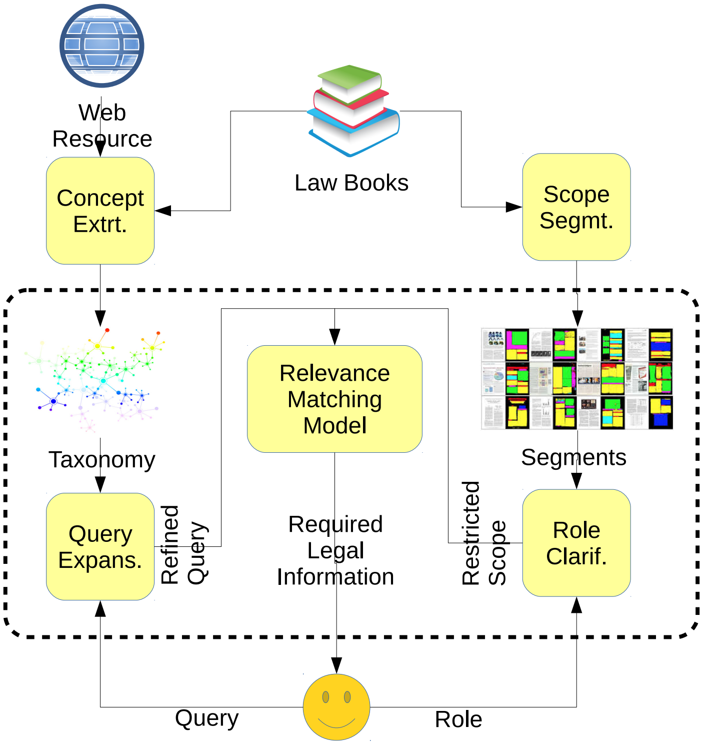
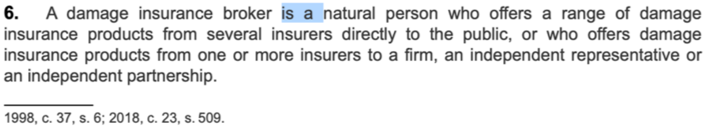


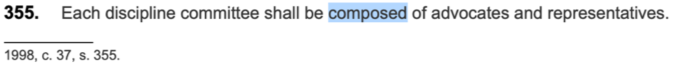
Figure 2.3: The framework for legal information retrieval.



6. A damage insurance broker is a natural person who offers a range of damage insurance products from several insurers directly to the public, or who offers damage insurance products from one or more insurers to a firm, an independent representative or an independent partnership.

1998, c. 37, s. 6; 2018, c. 23, s. 509.

Figure 2.4: An example of the concepts in an "IS-A" relation.



355. Each discipline committee shall be composed of advocates and representatives.

1998, c. 37, s. 355.

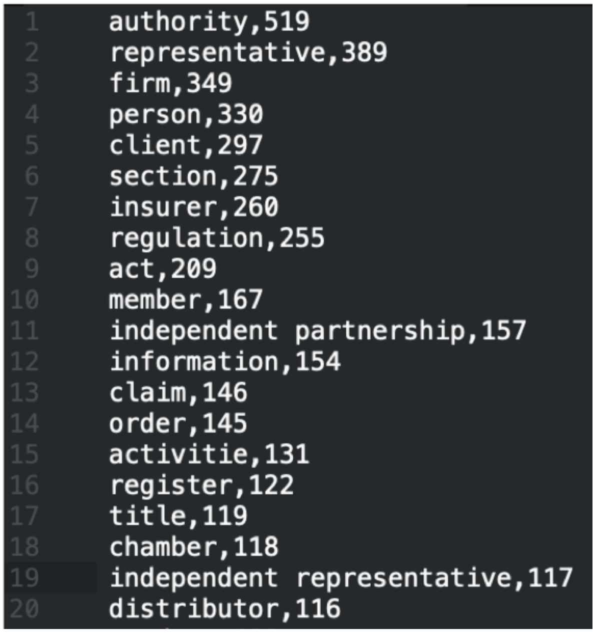
Figure 2.5: An example of the concepts in a "PART-OF" relation.

Taxonomy results

As a result 378 concepts were extracted from the "Loi" documents and 418 concepts from all the documents included in Distribution of financial products and services. Some concepts included in the larger taxonomy are displayed in Figure 2.6, where the numbers are frequencies.

2.4.2 Related terms extraction

Query expansion aims at tackling the problem of term mismatching: it expands the original query based on relations between terms. Two distinct types of relations can be used for query expansion. The first type is a relation between the terms within the taxonomy in the legal area, for instance the relation between the term "independent" and "partnership" or "representative." The second type is a relation between a general query term and its related concepts in the legal documents. Such a relation can be extracted using co-occurrences. For example the query term "injury" may be associated to "damage insurance product." The first type of relation can be used for expanding the query



```
authority,519
representative,389
firm,349
person,330
client,297
section,275
insurer,260
regulation,255
act,209
member,167
independent partnership,157
information,154
claim,146
order,145
activitie,131
register,122
title,119
chamber,118
independent representative,117
distributor,116
```

Figure 2.6: A sample of the concepts extracted for the construction of the taxonomy.

terms submitted by professional users, while the second type of relations are more appropriate for non-professional users. Therefore related terms extraction is conducted respectively for professional users and non-professional users.

For professionals

For professional users, the rationale is that they are more familiar with the terminology than non-professional users. Even if the submitted query uses the correct terms in the taxonomy, however, the results may be improved by expanding the query with additional related concepts found in the legal documents. Hence the taxonomy to be extracted for professionals is constructed using the legal documents.

Relatedness The relatedness is typically measured by various similarity metrics, such as the Jaccard similarity, the Dice similarity, the Cosine similarity, the Normalized mutual information, and so on. We use x and y to denote terms, D denotes a document set of size N, D_x the subset of documents containing the term x, and df_{xy} the number of co-occurences of the two terms x and y, i.e., the cardinality of $D_x \cap D_y$. The similarity metrics are defined as follows.

- Jaccard Similarity

$$S_1(x,y) = |D_x \cap D_y| / |D_x \cup D_y| = df_{xy} / \left(df_x + df_y - df_{xy}\right)$$

- Dice Similarity

$$S_2(x,y) = 2|D_x \cap D_y| / \left(|D_x| + |D_y|\right) = 2df_{xy} / \left(df_x + df_y\right)$$

- Cosine Similarity

$$S_3(x,y) = |D_x \cap D_y| / \sqrt{|D_x| \cdot |D_y|} = df_{xy} / \sqrt{df_x \cdot df_y}$$

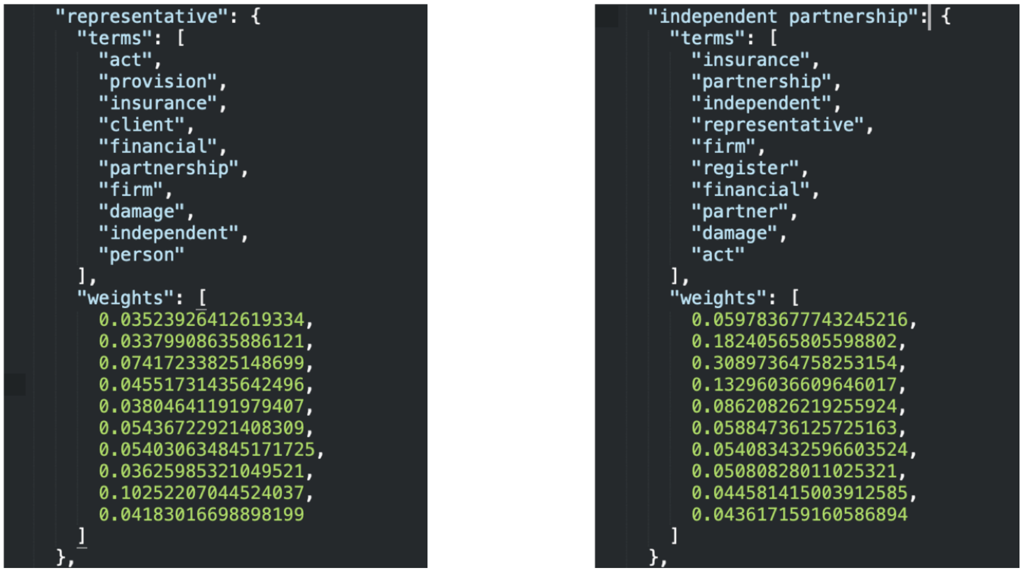
- Normalized mutual information

$$S_4(x,y) = \left(\log \frac{P(x,y)}{P(x)P(y)}\right) / \log N,$$

where $P(x)$ denotes df_x/N, $P(y)$ denotes df_y/N, and $P(x,y)$ denotes df_{xy}/N.

According to Vechtomova [1] and Myoung et al. [2], Cosine similarity and Jaccard similarity usually lead to better query expansion results. In this work we adopt Cosine similarity for measuring relatedness.

Results We extracted 10 related terms for each concept in the taxonomy. Some examples are shown in Figure 2.7(a) and Figure 2.7(b).



```
"representative": {
  "terms": [
    "act",
    "provision",
    "insurance",
    "client",
    "financial",
    "partnership",
    "firm",
    "damage",
    "independent",
    "person"
  ],
  "weights": [
    0.03523926412619334,
    0.03379908635886121,
    0.07417233825148699,
    0.04551731435642496,
    0.03804641191979407,
    0.05436722921408309,
    0.054030634845171725,
    0.03625985321049521,
    0.10252207044524037,
    0.04183016698898199
  ]
},
```

(a) Terms related to “representative”.

```
"independent partnership": {
  "terms": [
    "insurance",
    "partnership",
    "independent",
    "representative",
    "firm",
    "register",
    "financial",
    "partner",
    "damage",
    "act"
  ],
  "weights": [
    0.059783677743245216,
    0.18240565805598802,
    0.30897364758253154,
    0.13296036609646017,
    0.08620826219255924,
    0.05884736125725163,
    0.054083432596603524,
    0.05080828011025321,
    0.044581415003912585,
    0.043617159160586894
  ]
},
```

(b) Terms related to “independent partnership”.

Figure 2.7: Examples of the top-10 terms related to the query concept.

In the figure the numbers indicate how strongly each term is associated to the target concept in the legal materials.

For non-professionals

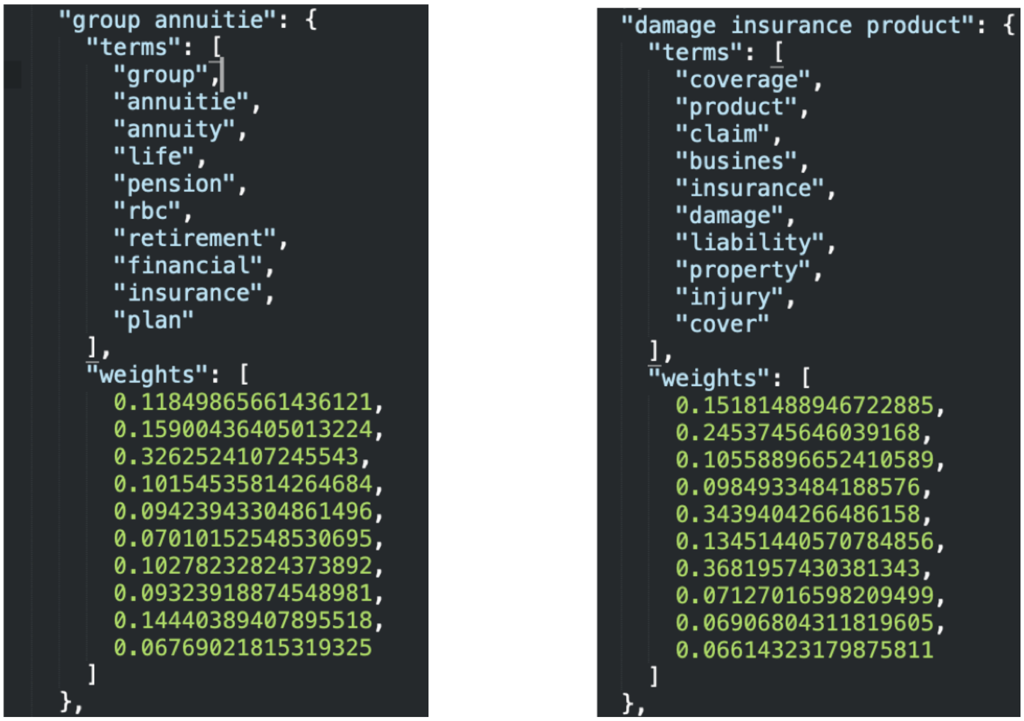
For non-professional users, our intuition is that they tend to use more “daily life” terms when searching legal materials than a professional would. If we know that one general term (in the query) is strongly associated with a concept in the taxonomy, expanding the query term with the taxonomy concept is probably helpful.

We used an external search engine to detect the association between general terms and taxonomy terms. We take each concept in the taxonomy as a query term submitted to Micosoft Bing (`https://www.bing.com/`); the top-ranked results are taken as the document resources to build the associations between the concept and any other general terms. The TF-IDF index is used to measure the relatedness and importance of each term to the query concept.

Examples of the top-10 strongly related general terms extracted from the related terms set are shown in Figure 2.8(a) and Figure 2.8(b). Each number indicate the strength of the association between the query concept and the general terms in the relevant documents returned by the search engine.

2.4.3 Query expansion

Query expansion aims to enrich the user’s query by adding additional search terms, either automatically or interactively. The added terms may help represent the user’s information needs more accurately and completely, thus increasing the chance of matching the user’s query to the relevant documents.



```
  "group annuitie": {
    "terms": [
      "group",
      "annuitie",
      "annuity",
      "life",
      "pension",
      "rbc",
      "retirement",
      "financial",
      "insurance",
      "plan"
    ],
    "weights": [
      0.11849865661436121,
      0.15900436405013224,
      0.3262524107245543,
      0.10154535814264684,
      0.09423943304861496,
      0.07010152548530695,
      0.10278232824373892,
      0.09323918874548981,
      0.14440389407895518,
      0.06769021815319325
    ]
  },
```

(a) Terms related to “group annuity”.

```
  "damage insurance product": {
    "terms": [
      "coverage",
      "product",
      "claim",
      "busines",
      "insurance",
      "damage",
      "liability",
      "property",
      "injury",
      "cover"
    ],
    "weights": [
      0.15181488946722885,
      0.2453745646039168,
      0.10558896652410589,
      0.0984933484188576,
      0.3439404266486158,
      0.13451440570784856,
      0.3681957430381343,
      0.07127016598209499,
      0.06906804311819605,
      0.06614323179875811
    ]
  },
```

(b) Terms related to “damage insurance product”.

Figure 2.8: Examples of the top-10 terms related to the query concept.

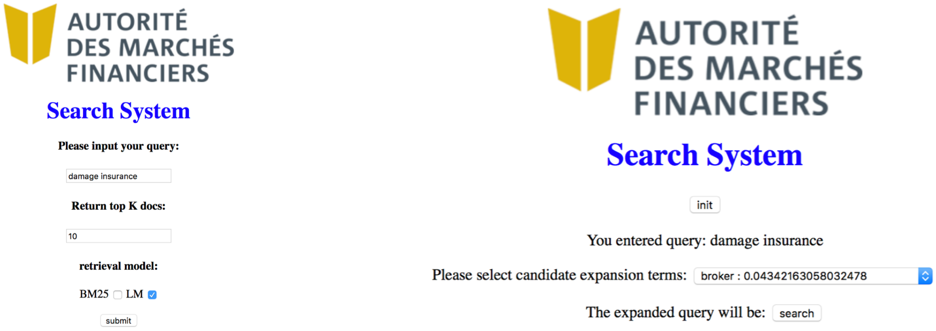
Query expansion can be performed automatically or interactively. In automatic query expansion (AQE) [3, 4, 5], the system selects and adds terms directly to the user’s query, whereas in interactive query expansion (IQE) [6, 7, 8], the system selects candidate terms for query expansion, shows them to the user, and asks the user to select (or deselect) terms that he (she) wants to include into (or exclude from) the query.

According to the resources they use for query expansion [9], the methods can be divided into the following categories.

- **Relevance Feedback** [10] QE terms are extracted from the documents retrieved in response to the user’s query and judged relevant by the user.
- **Pseudo-Relevance Feedback** [11] QE terms are extracted from the top-ranked documents retrieved in response to the user’s query.
- **Association Thesauri** QE uses association thesauri that are built automatically and collection-wide word co-occurrences.

Since we have acquired a taxonomy and its associated concepts from the legal documents, and the general terms from open documents by searching the web, our query expansion method falls into the third category - association thesauri.

Unlike the categories using relevance or pseudo-relevance feedback (where terms are selected from documents at search time), QE techniques in the third category rely on lexical resources constructed automatically prior to the search process. Statistical measures of term similarity are typically used to identify terms in a large document collection that co-occur in the same contexts, and therefore, are likely to be conceptually related. For example, Qiu and Frei [12] developed a query expansion method where query expansion terms are selected from a co-occurrence based term-term similarity thesaurus (built automatically) on the basis of the degree of their similarity to all terms in the query. Jing and Croft [13] developed a technique for automatic construction of a co-occurrence thesaurus.


AUTORITÉ DES MARCHÉS FINANCIERS
Search System
submit
AUTORITÉ DES MARCHÉS FINANCIERS
Search System
init
add
search

Please input your query: damage insurance

Return top K docs: 10

retrieval model: BM25 □ LM ☑

You entered query: damage insurance

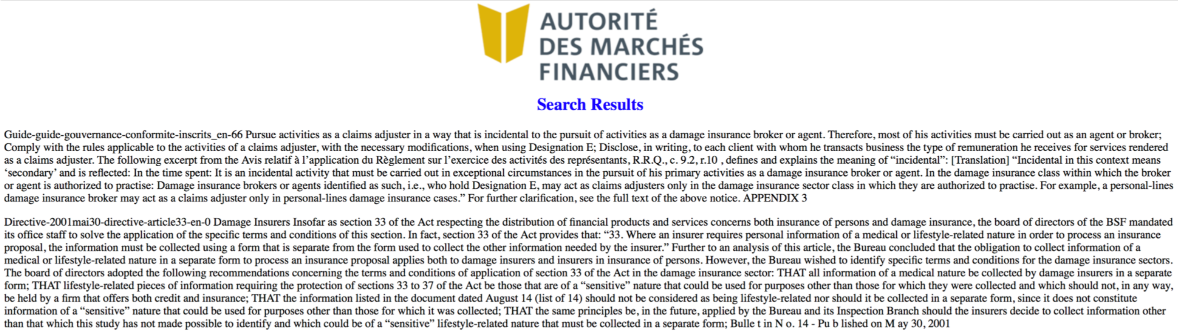
Please select candidate expansion terms: broker : 0.04342163058032478

The expanded query will be:

(a) A query input.

(b) Query expansion for the input term.

Figure 2.9: An example of query expansion.



Search Results

Guide-guide-gouvernance-conformite-inscrits_en-66 Pursue activities as a claims adjuster in a way that is incidental to the pursuit of activities as a damage insurance broker or agent. Therefore, most of his activities must be carried out as an agent or broker; Comply with the rules applicable to the activities of a claims adjuster, with the necessary modifications, when using Designation E; Disclose, in writing, to each client with whom he transacts business the type of remuneration he receives for services rendered as a claims adjuster. The following excerpt from the Avis relatif à l'application du Règlement sur l'exercice des activités des représentants, R.R.Q., c. 9.2, r.10 , defines and explains the meaning of "incidental": [Translation] "Incidental in this context means 'secondary' and is reflected: In the time spent: It is an incidental activity that must be carried out in exceptional circumstances in the pursuit of his primary activities as a damage insurance broker or agent. In the damage insurance class within which the broker or agent is authorized to practise: Damage insurance brokers or agents identified as such, i.e., who hold Designation E, may act as claims adjusters only in the damage insurance sector class in which they are authorized to practise. For example, a personal-lines damage insurance broker may act as a claims adjuster only in personal-lines damage insurance cases." For further clarification, see the full text of the above notice. APPENDIX 3

Directive-2001mai30-directive-article33-en-0 Damage Insurers Insofar as section 33 of the Act respecting the distribution of financial products and services concerns both insurance of persons and damage insurance, the board of directors of the BSF mandated its office staff to solve the application of the specific terms and conditions of this section. In fact, section 33 of the Act provides that: "33. Where an insurer requires personal information of a medical or lifestyle-related nature in order to process an insurance proposal, the information must be collected using a form that is separate from the form used to collect the other information needed by the insurer." Further to an analysis of this article, the Bureau concluded that the obligation to collect information of a medical or lifestyle-related nature in a separate form to process an insurance proposal applies both to damage insurers and insurers in insurance of persons. However, the Bureau wished to identify specific terms and conditions for the damage insurance sectors. The board of directors adopted the following recommendations concerning the terms and conditions of application of section 33 of the Act in the damage insurance sector: THAT all information of a medical nature be collected by damage insurers in a separate form; THAT lifestyle-related pieces of information requiring the protection of sections 33 to 37 of the Act be those that are of a "sensitive" nature that could be used for purposes other than those for which they were collected and which should not, in any way, be held by a firm that offers both credit and insurance; THAT the information listed in the document dated August 14 (list of 14) should not be considered as being lifestyle-related nor should it be collected in a separate form, since it does not constitute information of a "sensitive" nature that could be used for purposes other than those for which it was collected; THAT the same principles be, in the future, applied by the Bureau and its Inspection Branch should the insurers decide to collect information other than that which this study has not made possible to identify and which could be of a "sensitive" lifestyle-related nature that must be collected in a separate form; Bulle t in N o. 14 - Pu b lished on M ay 30, 2001

Figure 2.10: The search results for the expanded query terms.

Given the concepts and their strengths of relation to the query term, it is quite straightforward to expand the query in an interactive way. A typical use case of query expansion in an interactive way when searching for legal information is shown in Figure 2.9(a) and Figure 2.9(b).

When processing a query, e.g., "damage insurance," the search engine will provide, for the expansion, a set of terms with weights indicating the strength of the association. With the selected terms, e.g., "broker" with association weight of 0.043 in the present case, the search engine will return the results related to the expanded query. The top two results for the term "damage insurance" expanded by "broker" are shown in Figure 2.10.

2.4.4 Scope clarification

In legal information retrieval, a user may be associated with a specific role when searching for documents. Certain sections of the rule books may be relevant (or not), even though they are about the required concept. For example, regulations for a damage insurance broker and a financial planner could be different. By restricting the search scope within the sections of interest, we can improve the efficiency of the information retrieval.

To return more accurate results to clients with specific roles, our chosen solution is scope search, which requires us to segment in advance the corpus according to the interests of different roles or organizations.

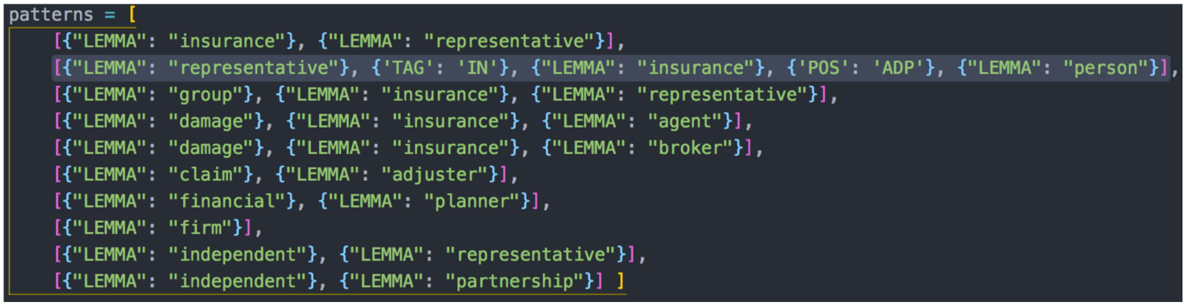
Roles

The roles and related scopes are listed below. We consider three types of employees: insurance representatives, claims adjusters, and financial planners. The three types for an employer are, respectively: firm, independent representative, and independent partnership.

- Employee
  - Insurance Representative
    - Representative in insurance of person
    - Group insurance representative
    - Damage insurance agent
    - Damage insurance broker
  - Claims Adjuster
  - Financial Planner
- Employer
  - Firm
  - Independent Representative
  - Independent Partnership

Scope segmentation

We adopted a rule-based solution for scope segmentation. A set of rules are designed for each role. An example of rules is shown in Figure 2.11.



```
patterns = [
    [{"LEMMA": "insurance"}, {"LEMMA": "representative"}],
    [{"LEMMA": "representative"}, {'TAG': 'IN'}, {"LEMMA": "insurance"}, {'POS': 'ADP'}, {"LEMMA": "person"}],
    [{"LEMMA": "group"}, {"LEMMA": "insurance"}, {"LEMMA": "representative"}],
    [{"LEMMA": "damage"}, {"LEMMA": "insurance"}, {"LEMMA": "agent"}],
    [{"LEMMA": "damage"}, {"LEMMA": "insurance"}, {"LEMMA": "broker"}],
    [{"LEMMA": "claim"}, {"LEMMA": "adjuster"}],
    [{"LEMMA": "financial"}, {"LEMMA": "planner"}],
    [{"LEMMA": "firm"}],
    [{"LEMMA": "independent"}, {"LEMMA": "representative"}],
    [{"LEMMA": "independent"}, {"LEMMA": "partnership"}] ]
```

Figure 2.11: An example of rules for scope segmentation.

Rule matching functions are implemented with the assistance of the open source NLP toolset spaCY (`https://spacy.io/`). As shown in the previous section, the roles are organized in a hierarchical structure: hence the scopes are also distributed into the corresponding structure. When trying to insert a section into the scope hierarchy, if the child section doesn't match a scope role, it will inherit the role of the parent.

2.4.5 Retrieval models

The kernel part of a search engine is its relevance matching model. We employed the popular BM25 [14] and the language model (LM) [20] to perform retrieval. The BM25 model relies on the term frequencies and inverse document frequencies of the query terms appearing in documents. Given a query $q = [q_1, \ldots, q_n]$ and a document $d = [d_1, \ldots, d_m]$, the relevance score is determined by Equation (2.1):

$$S(q,d) = \sum_{i=1}^{n} IDF(q_i) \frac{f(q_i,d)(k_1+1)}{f(q_i,d) + k_1(1 - b + b\frac{|D|}{avdl})}, \tag{2.1}$$

where q_i is the i_{th} query term, d is the document, $IDF(q_i)$ is the inverse document frequency for query term q_i, $f(q_i,d)$ is the term frequency of query term q_i appearing in document d, $|D|$ is the number of documents in the whole collection, $avdl$ is the average document length of the collection, and k_1 and b are parameters of the model.

The LM retrieval model relies on both the frequency of the exactly matched query term in the document and the frequency of the term in the whole collection. It employs a language model with smoothing to estimate the relevance of query q with respect to document d. The relevance score is determined by Equation (2.2):

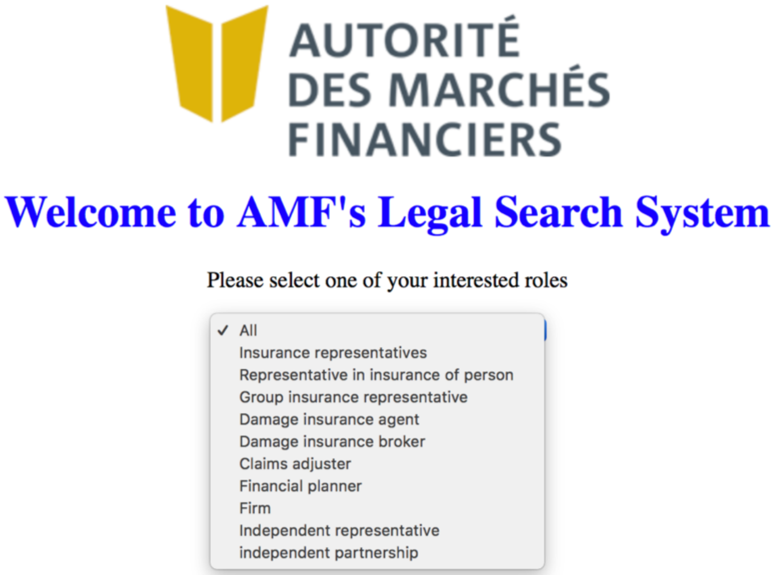
$$P(w|d) = \frac{c(w,d) + \mu P(w|C)}{\sum_w c(w,d) + \mu}, \tag{2.2}$$

where $P(w|d)$ is the estimated relevance contribution of query term w, $c(w,d)$ is the frequency of query term w in document d, $P(w|C)$ is the probability of query term w occurring in the background/collection, and μ is the Dirichlet smoothing parameter.

Finally the legal document retrieval framework is implemented through a popular open source tool called Indri `https://sourceforge.net/projects/lemur/files/lemur/indri-5.3/`.

2.4.6 A legal document search interface

When a user wishes to find the relevant sections within legal documents, he/she should select a specific role. The search will then be carried out in the corresponding scope only, as shown in Figure 2.12.



Welcome to AMF's Legal Search System

Please select one of your interested roles

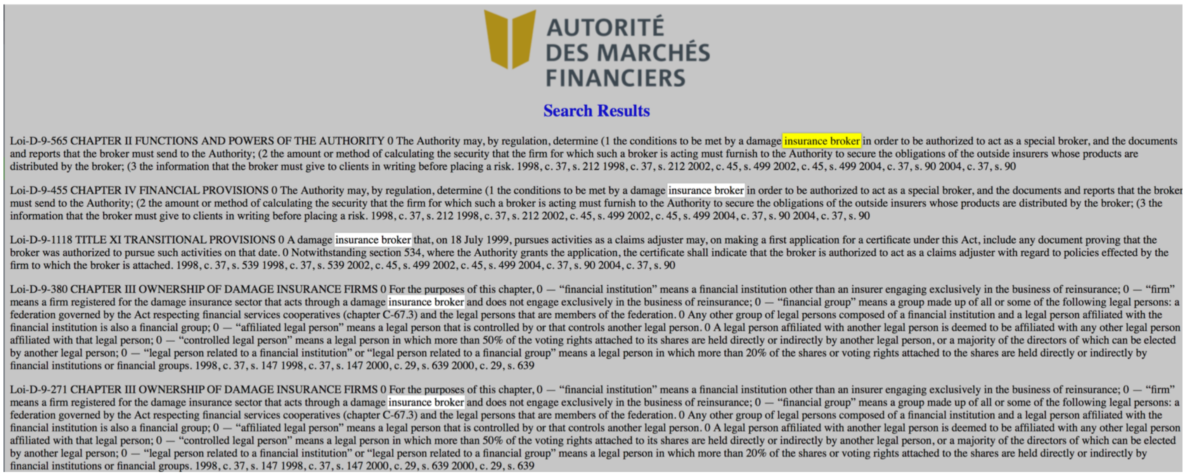
✓ All
Insurance representatives
Representative in insurance of person
Group insurance representative
Damage insurance agent
Damage insurance broker
Claims adjuster
Financial planner
Firm
Independent representative
independent partnership

Figure 2.12: Scope selection.

If the user chooses “all” as the role, the search will be conducted across all the documents in the collection without using a role to filter the results.

If the user has chosen a role, he or she will be redirected to a search interface where the user can enter a query and the desired number of documents to be returned. Also the user has the opportunity to select a specific retrieval model (BM25 or LM).

The user can also perform a query expansion with the taxonomy integrated into the system. On the welcoming page, if the user ticks the box “with query expansion,” the interface will show a list of candidate terms that are associated with the original query and their weights. The user can then choose the expansion terms from the drop-down list. Once the expanded query is submitted to the system, the system will search documents with the expanded query. An example of list returned by the legal document search engine is illustrated in Figure 2.13.



AUTORITÉ DES MARCHÉS FINANCIERS

Search Results

Loi-D-9-565 CHAPTER II FUNCTIONS AND POWERS OF THE AUTHORITY 0 The Authority may, by regulation, determine (1 the conditions to be met by a damage insurance broker in order to be authorized to act as a special broker, and the documents and reports that the broker must send to the Authority; (2 the amount or method of calculating the security that the firm for which such a broker is acting must furnish to the Authority to secure the obligations of the outside insurers whose products are distributed by the broker; (3 the information that the broker must give to clients in writing before placing a risk. 1998, c. 37, s. 212 1998, c. 37, s. 212 2002, c. 45, s. 499 2002, c. 45, s. 499 2004, c. 37, s. 90 2004, c. 37, s. 90

Loi-D-9-455 CHAPTER IV FINANCIAL PROVISIONS 0 The Authority may, by regulation, determine (1 the conditions to be met by a damage insurance broker in order to be authorized to act as a special broker, and the documents and reports that the broker must send to the Authority; (2 the amount or method of calculating the security that the firm for which such a broker is acting must furnish to the Authority to secure the obligations of the outside insurers whose products are distributed by the broker; (3 the information that the broker must give to clients in writing before placing a risk. 1998, c. 37, s. 212 1998, c. 37, s. 212 2002, c. 45, s. 499 2002, c. 45, s. 499 2004, c. 37, s. 90 2004, c. 37, s. 90

Loi-D-9-1118 TITLE XI TRANSITIONAL PROVISIONS 0 A damage insurance broker that, on 18 July 1999, pursues activities as a claims adjuster may, on making a first application for a certificate under this Act, include any document proving that the broker was authorized to pursue such activities on that date. 0 Notwithstanding section 534, where the Authority grants the application, the certificate shall indicate that the broker is authorized to act as a claims adjuster with regard to policies effected by the firm to which the broker is attached. 1998, c. 37, s. 539 1998, c. 37, s. 539 2002, c. 45, s. 499 2002, c. 45, s. 499 2004, c. 37, s. 90 2004, c. 37, s. 90

Loi-D-9-380 CHAPTER III OWNERSHIP OF DAMAGE INSURANCE FIRMS 0 For the purposes of this chapter, 0 — “financial institution” means a financial institution other than an insurer engaging exclusively in the business of reinsurance; 0 — “firm” means a firm registered for the damage insurance sector that acts through a damage insurance broker and does not engage exclusively in the business of reinsurance; 0 — “financial group” means a group made up of all or some of the following legal persons: a federation governed by the Act respecting financial services cooperatives (chapter C-67.3) and the legal persons that are members of the federation. 0 Any other group of legal persons composed of a financial institution and a legal person affiliated with the financial institution is also a financial group; 0 — “affiliated legal person” means a legal person that is controlled by or that controls another legal person. 0 A legal person affiliated with another legal person is deemed to be affiliated with any other legal person affiliated with that legal person; 0 — “controlled legal person” means a legal person in which more than 50% of the voting rights attached to its shares are held directly or indirectly by another legal person, or a majority of the directors of which can be elected by another legal person; 0 — “legal person related to a financial institution” or “legal person related to a financial group” means a legal person in which more than 20% of the shares or voting rights attached to the shares are held directly or indirectly by financial institutions or financial groups. 1998, c. 37, s. 147 1998, c. 37, s. 147 2000, c. 29, s. 639 2000, c. 29, s. 639

Loi-D-9-271 CHAPTER III OWNERSHIP OF DAMAGE INSURANCE FIRMS 0 For the purposes of this chapter, 0 — “financial institution” means a financial institution other than an insurer engaging exclusively in the business of reinsurance; 0 — “firm” means a firm registered for the damage insurance sector that acts through a damage insurance broker and does not engage exclusively in the business of reinsurance; 0 — “financial group” means a group made up of all or some of the following legal persons: a federation governed by the Act respecting financial services cooperatives (chapter C-67.3) and the legal persons that are members of the federation. 0 Any other group of legal persons composed of a financial institution and a legal person affiliated with the financial institution is also a financial group; 0 — “affiliated legal person” means a legal person that is controlled by or that controls another legal person. 0 A legal person affiliated with another legal person is deemed to be affiliated with any other legal person affiliated with that legal person; 0 — “controlled legal person” means a legal person in which more than 50% of the voting rights attached to its shares are held directly or indirectly by another legal person, or a majority of the directors of which can be elected by another legal person; 0 — “legal person related to a financial institution” or “legal person related to a financial group” means a legal person in which more than 20% of the shares or voting rights attached to the shares are held directly or indirectly by financial institutions or financial groups. 1998, c. 37, s. 147 1998, c. 37, s. 147 2000, c. 29, s. 639 2000, c. 29, s. 639

Figure 2.13: Query results with scope restriction and query expansion.

Each individual paragraph resulting from the search is within the scope of the role “damage insurance broker.”

2.5 Conclusion

The goal of the team was to explore ideas useful for legal document retrieval. We proposed a legal document retrieval framework to tackle the two major problems in accessing legal information: the term-mismatch problem and the role-dependent search. The framework has been implemented and tested and has demonstrated that the solutions could be useful in practice.

This work is of course limited because the workshop lasted for one week only. Three promising directions can be further explored.

1. It may be useful to exploit query logs for query expansion [15]. Such approaches have been found very useful in general search engines. We expect that the results could be largely improved through query logs since the query logs record the human behaviours that reveal which sections are truly relevant to the input query term. Thus query logs give a better indication of the association between terms and the relevant documents.
2. Question answering systems [16, 17] for legal problems could provide a more convenient solution to the users’ problem. Given that the law and regulations are formally defined and mandated, the structure of the language used in laws and regulations seems to be suitable for answering a question automatically (instead of locating the section where the answer is to be found). Besides the use of a QA system will lighten the regulatory burden more efficiently that an IR system.
3. Machine reading [18, 19] approaches for legal regulation understanding might help to decide whether a case complies with certain regulations or not: this is another recurring task for professionals and non-professionals in this field.

In conclusion, during the workshop we have been able to explore several useful ideas for legal information retrieval for the AMF clients. We have also identified several possible avenues for future research and developments.

Bibliography

[1] Olga Vechtomova. Query Expansion for Information Retrieval, pages 2254–2257. Springer US, Boston, MA, 2009.

[2] Myoung-Cheol Kim and Key-Sun Choi. A comparison of collocation-based similarity measures in query expansion. Information Processing & Management, 35(1):19–30, 1999.

[3] Hang Cui, Ji-Rong Wen, Jian-Yun Nie, and Wei-Ying Ma. Probabilistic query expansion using query logs. In Proceedings of the 11th international conference on World Wide Web, pages 325–332. ACM, 2002.

[4] Yogesh Gupta and Ashish Saini. A novel fuzzy-pso term weighting automatic query expansion approach using combined semantic filtering. Knowl.-Based Syst., 136:97–120, 2017.

[5] Jagendra Singh and Aditi Sharan. Rank fusion and semantic genetic notion based automatic query expansion model. Swarm and Evolutionary Computation, 38:295–308, 2018.

[6] Donna Harman. Towards interactive query expansion. In Proceedings of the 11th annual international ACM SIGIR conference on Research and development in information retrieval, pages 321–331. ACM, 1988.

[7] Donna Harman. Towards interactive query expansion. SIGIR Forum, 51(2):79–89, 2017.

[8] H. Bast, Debapriyo Majumdar, and Ingmar Weber. Efficient interactive query expansion with complete search. In Proceedings of the Sixteenth ACM Conference on Information and Knowledge Management, CIKM 2007, Lisbon, Portugal, November 6–10, 2007, pages 857–860, 2007.

[9] K Sparck Jones, Steve Walker, and Stephen E. Robertson. A probabilistic model of information retrieval: development and comparative experiments: Part 2. Information processing & management, 36(6):809–840, 2000.

[10] Sicong Zhang, Dongyi Guan, and Hui Yang. Query change as relevance feedback in session search. In The 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2013, Dublin, Ireland, July 28–August 01, 2013, pages 821–824, 2013.

[11] Ali Montazeralghaem, Hamed Zamani, and Azadeh Shakery. Theoretical analysis of interdependent constraints in pseudo-relevance feedback. In The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08–12, 2018, pages 1249–1252, 2018.

[12] Yonggang Qiu and Hans-Peter Frei. Concept based query expansion. In Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval, pages 160–169. ACM, 1993.

[13] Yufeng Jing and W Bruce Croft. An association thesaurus for information retrieval. In Intelligent Multimedia Information Retrieval Systems and Management–Volume 1, pages 146–160. Le centre de hautes études internationales d'informatique documentaire, 1994.

[14] Stephen E. Robertson and Hugo Zaragoza. The probabilistic relevance framework: BM25 and beyond. Foundations and Trends in Information Retrieval, 3(4):333–389, 2009.

[15] Wei Gao, Cheng Niu, Jian-Yun Nie, Ming Zhou, Kam-Fai Wong, and Hsiao-Wuen Hon. Exploiting query logs for cross-lingual query suggestions. ACM Trans. Inf. Syst., 28(2):6:1–6:33, 2010.

[16] Gayle McElvain, George Sanchez, Don Teo, and Tonya Custis. Non-factoid question answering in the legal domain. In Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21–25, 2019, pages 1395–1396, 2019.

[17] Gayle McElvain, George Sanchez, Seán Matthews, Don Teo, Filippo Pompili, and Tonya Custis. Westsearch plus: A non-factoid question-answering system for the legal domain. In Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21–25, 2019., pages 1361–1364, 2019.

[18] Zhaohui Li, Yue Feng, Jun Xu, Jiafeng Guo, Yanyan Lan, and Xueqi Cheng. Teaching machines to extract main content for machine reading comprehension. In The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27–February 1, 2019, pages 9973–9974, 2019.

[19] Minghao Hu, Furu Wei, Yuxing Peng, Zhen Huang, Nan Yang, and Dongsheng Li. Read + verify: Machine reading comprehension with unanswerable questions. In The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27–February 1, 2019., pages 6529–6537, 2019.

[20] Chengxiang Zhai and John Lafferty. A Study of Smoothing Methods for Language Models Applied to Ad Hoc Information Retrieval IN SIGIR '01: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, September 2001., pages 334–342, 2001.

3 Constrained demand – Air Canada

Caroline Dietrich[a]
Fabian Bastin[b]
Mohammad Daneshvar[c]
Chiyu Ma[d]
Daniel Sallier[e]
Loïc Shi-Garrier[f]

[a] *Air Canada, Montréal (Québec), Canada*

[b] *Université de Montréal, Montréal (Québec), Canada*

[c] *Université du Québec à Montreal, Montréal (Québec), Canada*

[d] *Nankai University, Tianjin, China*

[e] *Saéro, Montréal (Québec), Canada*

[f] *École Nationale d'Aviation Civile, Toulouse, France*

October 2020
Les Cahiers du GERAD

3.1 Introduction

Air Canada is an international airline company operating more than 800 flights per day around the world. Some flight routes stay within Canada while other flights have an origin or destination located abroad. For each flight the booking window begins 354 days before the departure date. To increase its revenue the company needs to anticipate the demand for each flight prior to its departure, as this will allow the company to adapt the fares during the booking window. Since this information does not exist the company must forecast the demand, based on the booking information for previous flights of the company. The company evaluates the demand for each flight with the *Load Factor* parameter, which represents the portion of booked seats on the flight.

$$\text{Load Factor} \overset{\text{def}}{=} \frac{\text{number of booked seats on the departure day}}{\text{adjusted capacity on the departure day}}$$

The goal of this project is to predict the Load Factor at each day of the booking window. There are various issues to address. One of the main challenges in forecasting demand is the data, as the stored data captures the number of successful bookings only. Hence if a flight is fully booked, there is no data about the number of potential customers that were not able to book seats. Moreover, if the price is higher than what the passengers are willing to pay, they might not purchase a ticket; this data (on missed passengers) is unavailable, however.

3.2 Data

Air Canada provided us with flights booking data for some flight numbers and two years, i.e., 2017 and 2018. Each record indicates the flight number, the origin and destination names, the date of departure, the cabin type, whether it is a group booking or not, the price of the ticket, etc.

There are 60,229,667 data records in the dataset. Each of these records contains information about the number of bookings for a specific flight. The booking records are classified according to some features including cabin type, group or individual booking, days before the flight, etc. Hence for each flight the booking window includes multiple records per day. In some cases the information about a flight might change. It could be a change in the departure time, the flight number, or the cabin capacity (if the aircraft type changes). In addition to this challenge some records have missing values. This problem can occur even in important fields, such as the flight number.

For the tuning of model parameters and performance testing, we divided the data into three non-overlapping subsets: the training subset, the validation subset, and the testing subset. We included 60% of data in the training subset, 20% percent in the validation subset, and 20% in the testing subset.

3.3 Solution method

In the limited time we had during the IPSW, we tried to implement one of our ideas to solve this problem. Our idea was to use the booking information and load factor of previous flights as a model to forecast the load factor of future flights. To explain the solution method we need to introduce the concept of booking sequence.

3.3.1 Booking sequence

As mentioned before there is a booking window for each flight, where potential passengers can book the flight seats. We count the number of bookings per flight-cabin for each day of the booking window, starting 354 days before the departure date. We then compute the cumulative number of booked seats for each day of the booking window and build the Booking Sequence of the flight-cabin.

Since we are going to compare these sequences and the cabin capacities vary from aircraft to aircraft and thus from flight to flight, we normalize the booking sequence, i.e., divide its values by the adjusted capacity of the cabin at the departure date. We expect the maximum number in each sequence to be 1.0 but there are sequences whose maximum is as high as 1.4. Indeed the number of booked seats can be greater than the number of available seats and the overload will be carried into a higher-class cabin. Figure 3.1 displays the booking sequences of 500 flight-cabins. There are some group bookings at the beginning of the booking sequences. We have been told by Air Canada that most of these group bookings will be cancelled within two months after the booking. At the right end of the sequences one can observe loading factors with values greater than one (indicating overloading in the corresponding flight-cabins).

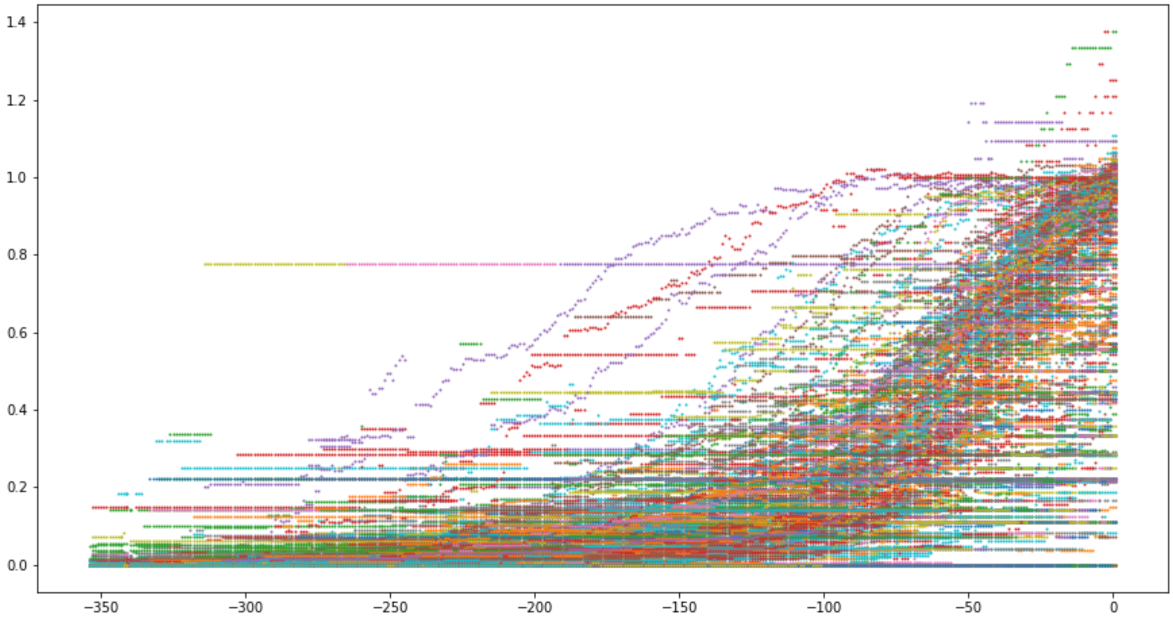


Figure 3.1: Booking sequences of 500 flight-cabins. The x-axis is the opposite of the number of days before the departure date and the y-axis is the load factor of the cabin.

As mentioned in the previous section we selected 60% of the data for the training subset (1996 booking sequences), 665 booking sequences for the validation subset, and 666 booking sequences (the remaining ones) for the testing subset.

3.3.2 Distance function

For the training part of our model we need a function expressing the distance between any two booking sequences. We chose the 2-norm distance function in this project. Assume we have two booking sequences $\overrightarrow{b_1} = (b_1^{-354}, b_1^{-353}, ..., b_1^0)$ and $\overrightarrow{b_2} = (b_2^{-354}, b_2^{-353}, ..., b_2^0)$. The distance between these booking sequences is defined as follows.

$$D_{\overrightarrow{b_1},\overrightarrow{b_2}} = \sqrt{\sum_{i=-354}^{0} (b_1^i - b_2^i)^2}$$

The number of bookings per day increases as the departure date becomes closer. Hence we considered including a discount factor in our distance function in order to give more weight to the bookings that are close to the departure date. The discounted distance is given by the following formula, where r denotes the discount rate.

$$\sqrt{\sum_{i=-354}^{0} e^{ri}(b_1^i - b_2^i)^2}$$

3.3.3 Neighbourhood

We define the neighbourhood of a booking sequence S as the set of booking sequences in the training subset whose distance to S is less than ϵ. Here ϵ is our neighbourhood parameter, which should be tuned.

3.3.4 Forecast Load Factor

We need a function to predict the final load factor, given the sequences in the neighbourhood (where the final load factor is defined as the load factor one day before the departure date). The prediction must be made for each day of the booking window. For each booking sequence S, for each day in its booking window, we find the sequences in its neighbourhood and compute the mean, the median, and the mode of the load factors of sequences in the neighbourhood of S (in order to estimate the final load factors).

3.4 Experimental result

To tune the problem parameters we run the model over the validation data to find the best values for the parameters, including ϵ and the discount rate r. The figure below displays the plots of the load factor errors over the validation data.

We used two values for ϵ to assess the impact of neighbourhood change on the final prediction. Figure 3.2 displays the error value over all sequences in the validation set with $\epsilon = 0.5$ and Figure 3.3 displays the error for the case $\epsilon = 0.25$. As can be seen in these two figures there is not a significant difference between the average errors in the two figures. The only difference concerns the maximum value of the error. As the neighbourhood becomes smaller the curve of the maximum error is smoother and the maximum error in the final days of the booking window is much smaller than in the case of the larger neighbourhood.

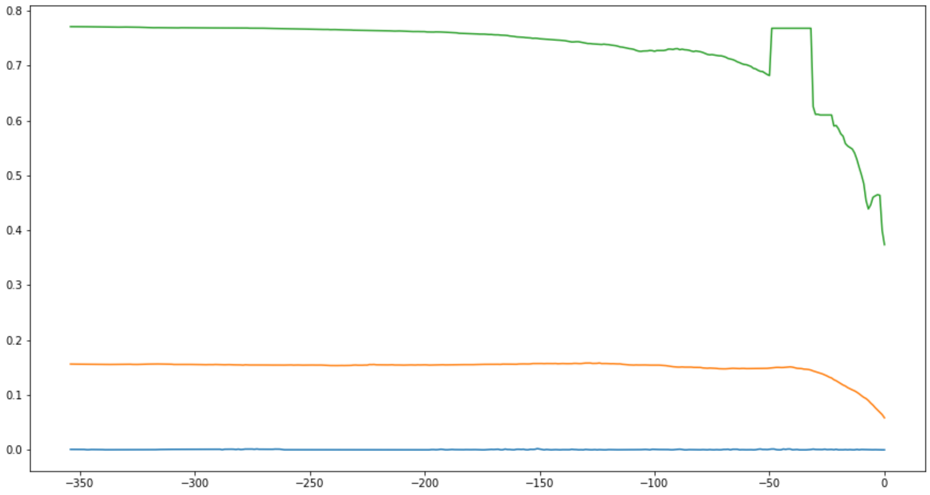


Figure 3.2: The average (orange), maximum (green), and minimum (blue) load factor error, over all the validation data for each day in the booking window with neighbourhood parameter $\epsilon = 0.5$ and the mean estimator.

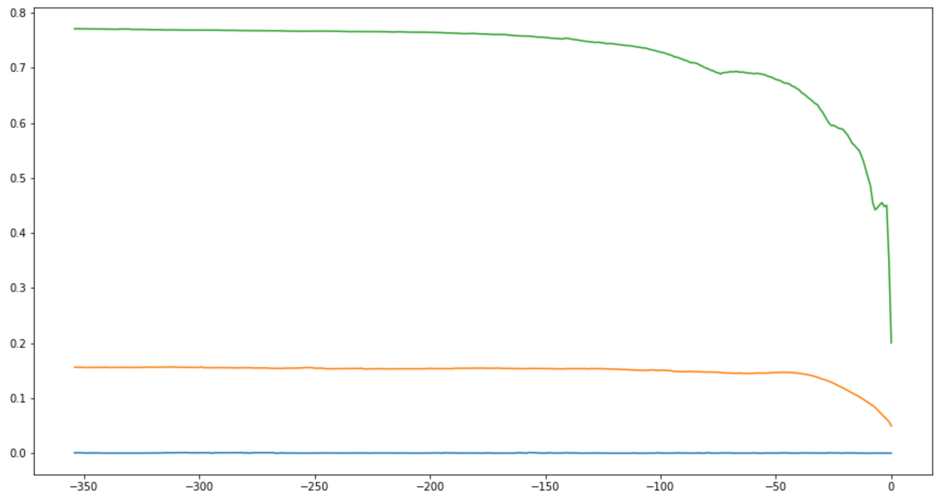


Figure 3.3: The average (orange), maximum (green), and minimum (blue) load factor error, over all the validation data for each day in the booking window with neighbourhood parameter $\epsilon = 0.25$ and the mean estimator.

On the basis of the above comparison we decided to set $\epsilon = 0.25$. To compare different estimator functions we ran the process using the mode and median estimators with $\epsilon = 0.25$. Figure 3.4 displays the results for the mode estimator and Figure 3.5 the results for the median estimator. The main difference between Figure 3.3, Figure 3.4, and Figure 3.5 concerns the maximum value of the error for each day of the booking window over the validation data. The difference between the average errors of the three estimators is small. Hence we decided to keep all three estimators instead of selecting a single one.

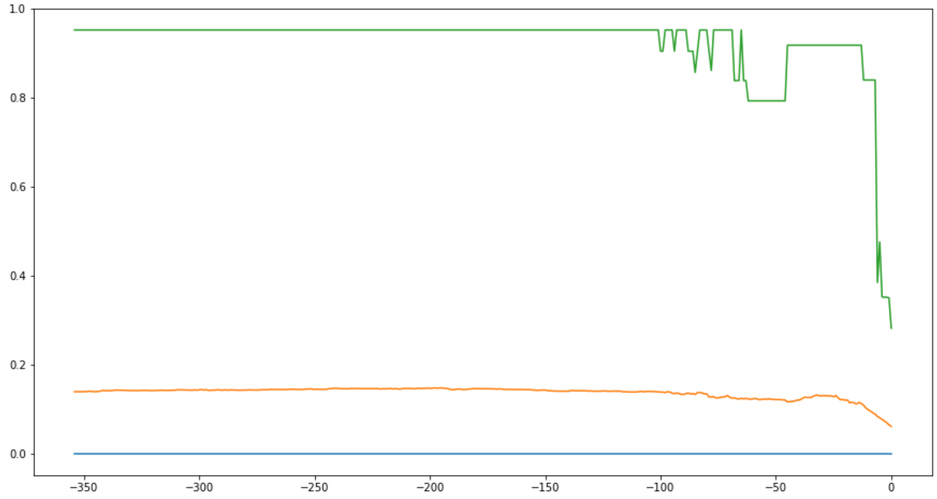


Figure 3.4: The average (orange), maximum (green), and minimum (blue) load factor error, over all the validation data for each day in the booking window with neighbourhood parameter $\epsilon = 0.25$ and the mode estimator.

After fixing the value of ϵ and comparing the estimator functions, we compared the results obtained for different values of the discount rate. The error in predicting the load factor on validation data, one day before departure, is presented in the table below. For every estimator the best value for the discount rate was found to be 0.05.

At this point all the model parameters had been set and we could run the model on the test data. The following table presents the results obtained by running the prediction models over the test data.

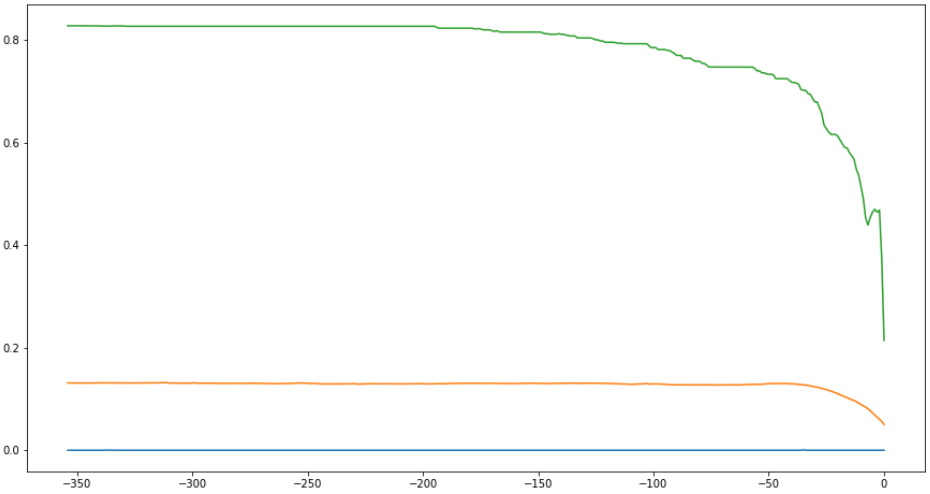


Figure 3.5: The average (orange), maximum (green), and minimum (blue) load factor error, over all the validation data for each day in the booking window with neighbourhood parameter $\epsilon = 0.25$ and the median estimator.

Table 3.1: Average difference between the predicted and real values of Load Factor one day before departure (validation data).

Discount Rate	Mean	Median	Mode
0.0	0.057	0.058	0.069
0.05	0.049	0.050	0.062
0.1	0.053	0.055	0.066

In these models the ϵ value is equal to 0.25 and the discount rate to 0.05. The median estimator function yields the best results but the difference between the errors of the three models is not large.

Table 3.2: Average difference between the predicted and real values of Load Factor one day before departure (test data).

Discount Rate	Mean	Median	Mode
0.05	0.064	0.062	0.069

To evaluate the model output more precisely it is useful to look at the distribution of the errors. We have discussed the average error for each day of the booking window but would like to know whether all sequences have error values close to the average or the error values have a large variance. Figure 3.6 displays the density of error values for the last 30 days of the booking window. We observe that most of the sequences have error values close to the average. The error can remain large, however, suggesting that noise remains and unexplained factors are difficult to capture. We also observe that the error tends to be smaller the closer one is from the departure date, as expected.

3.5 Future work

To continue this project we make the following suggestions.

- Filter the outliers.
- Use weighted load factor predictions: during the IPSW we considered three prediction models, based respectively on the mean, the mode, and the median. Our models have assumed that all the sequences in the neighbourhood have the same weight. In future studies one could consider carrying out the estimation with different weights for sequences in the neighbourhood.

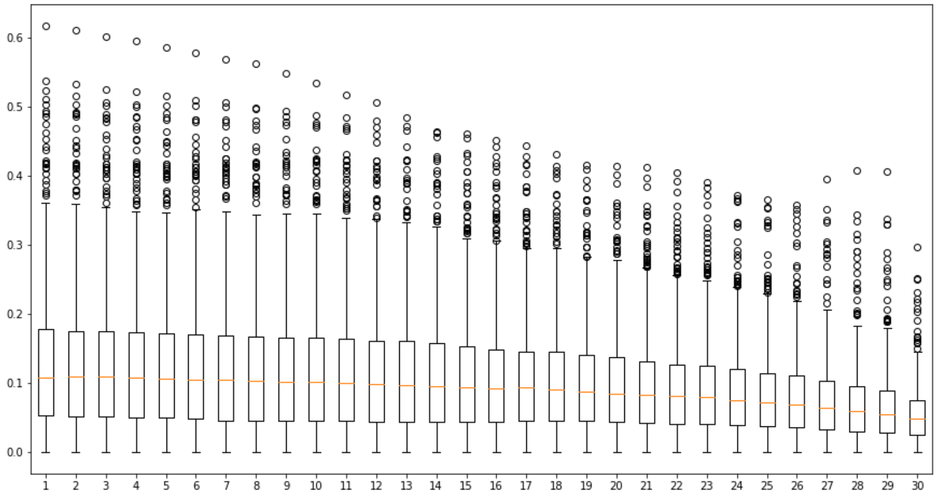


Figure 3.6: The boxplot of the model with the mean error estimator and with ϵ equal to 0.25 for the last 30 days of the booking window. The y-axis represents the error in forecasting the load factor and the x-axis is the index of the last 30 days of the booking window, with one as the day furthest from the departure date.

- Use other distance functions: we used the 2-norm as the distance function between two booking sequences. In future studies one could consider the impact of other distance functions on the forecast. We suggest investigating the ∞-norm and Frechet distances.
- Optimize model parameters such as the discount factor: given the time constraints of our project, we were not able to optimize the model parameters and our tuning did not necessarily yield the best value for each model parameter. We suggest spending more time finding the best values for each parameter of the model.
- Use regression techniques to detect similarity between shapes.

4 Flight spill detection

Ismael Assani [a]
Poclaire Kenmogne [a]
Gabriel Lemyre [a]
Thi Thanh Hue Nguyen [a]
François Bellavance [b]
Jiliang Li [c]
Frédérique Robin [d]
Pierre-Loïk Rothé [e]

[a] *Université de Montréal (Québec), Canada*

[b] *GERAD & HEC Montréal, Montréal (Québec), Canada*

[c] *University of Western Ontario, London (Ontario), Canada*

[d] *INRIA Saclay, Palaiseau, France*

[e] *ENPC, Marne-la-Vallée, France*

October 2020
Les Cahiers du GERAD

4.1 Introduction

The basic principle of revenue management is to try, as much as possible, to sell the right seat to the right passenger, at the right price and at the right time. The objective is to maximize the revenue provided by each seat sold and the cabin as a whole.

Since it is impossible to know with certainty the amount each passenger is willing to pay, airlines put a lot of effort into developing complex demand forecasting models. These forecasts are then used to assign a value to each seat via a network optimization model. This control value (for each seat) ultimately determines the price of each seat offered by the company. These prices vary as time unfolds and forecast demand becomes actual demand.

To enhance revenue management at Air Canada, our objective during the IPSW was to propose a predictor of a given flight spilling: in the next subsection we explain what "spilling" means. Later we present the dataset and describe the three main approaches we considered to solve the problem.

4.1.1 Flight spill: problem description

One of the key performance indicators used in the airline industry is the *spill*. A flight is in a "spill situation" if all its seats have been sold prior to departure. Such a situation raises concerns because close-in demand is generally of high value and seats must be saved for customers willing to buy seats a short time before departure. In some cases all the seats on a flight have been sold several days or even several weeks before departure. This may entail a lot of unrealized revenue and even dissatisfied customers trying to buy tickets a short time before departure. Often spill situations for a given flight are due to an inaccurate forecast of the demand for that flight. This question is crucial for airlines and has thus been tackled in the literature (see for instance [1]).

Since new information on demand and prices is added periodically to the airlines' database, the spill probability must be assessed on a regular basis during the "life" of the flight in order to update models and control demand. The proposed problem was to find a way to assess such probabilities and predict the time of first spill at all points during the "life" of a flight.

Because of time limitations we narrowed this objective to make it simpler and its solution easier to implement; our ultimate goal was to expand our work to the real-world situation. We chose to focus, in most approaches, on the probability that given all the information from the initial offering up to 30 days prior to the departure date, the flight will be in a spill situation three days prior to departure.

4.2 Dataset

4.2.1 About the data

We have two years of longitudinal measurements (i.e., bookings) for ten origin-destination pairs. The measurements are taken at regular time intervals between the day $-D$ (where D is at most 356) with respect to departure **and** the day of departure. Each flight is characterized by specific features (normalized city of departure, normalized destination, for instance), and is thus "unique." Some of these features are intrinsic to the aircraft (airplane capacity, etc.) while others are extrinsic (month of departure, etc.).

4.2.2 Preparing the data

We group flights by their unique flight index and approximate time of departure in order to link departures with different years but the same characteristics. We then choose to focus on one of these unique instances for which we have a lot of information in order to maximize the input of the models without making the problem too resource-consuming.

4.3 Proposed approaches

We explore several approaches, concentrating first on machine learning algorithms (random forests, lasso, SVM) and then on two modelling approaches: survival models and Kalman filtering.

4.3.1 Machine learning

One of the strengths of the machine learning approach is that it does not require“hard modelling” steps (i.e., writing a model with explicit dynamics) to make predictions (e.g., the weather of tomorrow). Indeed machine learning algorithms build a mathematical model based on sample data, known as “training data,” in order to make predictions or decisions without being explicitly programmed to perform the task.

In the sequel we use machine learning approaches to tackle two kinds of problems:

- classification problems: considering a new vector of observations composed of N features $x \in \mathbb{R}^N$, to which category $y \in \{0, 1\}$ does it belong? Here category 0 represents the event “no spill three days before departure” while category “1” is the event “spill three days before departure.”
- regression problems: features selection.

Random forests

Random forests denote a method for classification and regression that builds a multitude of decision trees on the training data set and outputs the class for each observation based on the majority votes of the decision trees. In our model a decision tree is a classifier using each feature as a binary decision till all features have been examined: then the method outputs the class. The random forest consists of a large number of trees where each tree is trained on the training data and then gives a prediction for the test data. The class prediction of the random forest is the majority of the trees’ predictions. In the case of our problem the random forest is used to predict whether or not the spill will happen three days prior to departure, and the proportion of trees that predict “spill = yes” is taken as the probability of the spill. One characteristic of the proportions is that they are mostly on the two ends of the range [0,1], since the trees rarely yield a half-half prediction. Because the company cares more about the result than the probability, it is reasonable to view the proportions as probabilities since they indicate more clearly to decision-makers whether there is going to be a spill or not. The importance of a feature in the random forest model is defined as the degree to which the node purity increases when carrying out a binary classificVM In Figure 4.1 we can observe that the “3-day prior-to-departure” loading factor from last year and the “30-day prior-to-departure” loading factor from the current year, together with the origin and destination of the flight, are the most important variables: this points to the need of a flight-specific analysis and modelling. Overall an 93% of accuracy was achieved using the random forest model.

Lasso

In statistics and machine learning, lasso (least absolute shrinkage and selection operator; also Lasso or LASSO) is a regression analysis method that performs both variable selection and regularization in order to enhance the prediction accuracy and interpretability of the statistical model it produces.

It was originally introduced (in 1986) in the geophysics literature (see [2]); in 1996 it was independently rediscovered and popularized by Robert Tibshirani [3], who coined the term and provided further insights into the observed performance. Lasso was originally formulated for least squares models

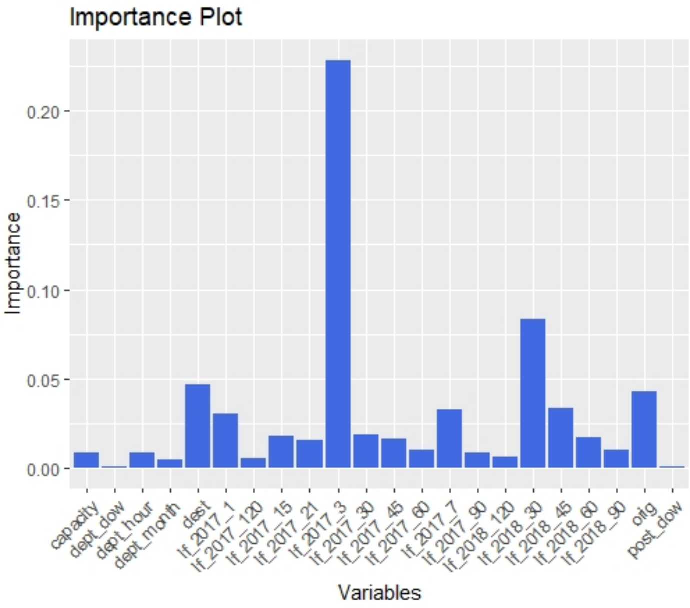


Figure 4.1: Importance plot.

and this simple case reveals a substantial amount about the behaviour of the estimator, including its relationship to ridge regression and best subset selection and the connections between lasso coefficient estimates and so-called soft thresholding. It also reveals that (like standard linear regression) the coefficient estimates do not need to be unique if covariates are collinear. We have a small dataset with at least 43 variables and use feature selection in order to identify the most important features. Here feature selection denotes the process of choosing a reduced number of explanatory variables to describe a response variable. Here are the main reasons why we are using feature selection.

- It makes the model easier to interpret, removing variables that are redundant and do not add any information.
- It reduces the size of the problem to enable algorithms to work faster, making it possible to handle high-dimensional data.
- It reduces overfitting.

The Lasso estimate is defined by the solution to the following l_1 optimization problem, where t is the upper bound on the sum of the coefficients.

$$\text{minimize} \left(\frac{\|Y - X\beta\|_2^2}{n} \right) \text{subject to} \sum_{j=1}^{k} \|\beta\|_1 < t \tag{4.1}$$

This optimization problem is equivalent to the following parameter estimation, where $\lambda \geq 0$ is the parameter that controls the strength of the penalty.

$$\hat{\beta}(\lambda) = \text{argmin}_{\beta} \left(\frac{\|Y - X\beta\|_2^2}{n} + \lambda \|\beta\|_1 \right) \tag{4.2}$$

The goal of our analysis is to determine which explanatory variables are most relevant when trying to predict the response (i.e., **spill**). In order to do so we will first analyze the dataset to gain a better understanding of the data. As we can see in Figure 4.2 some correlations between variables are stronger than others.

Table 4.1 displays the list of the most important factors, which we will use later in the classifying step.

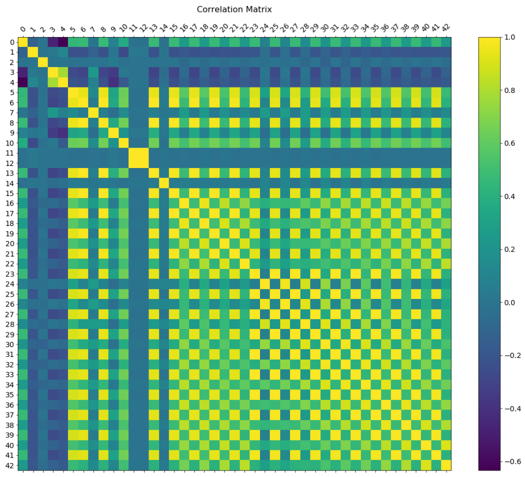


Figure 4.2: Matrix correlation.

Support vector machine (SVM)

SVM is an exciting algorithm and the concepts underlying it are relatively simple. The classifier separates data points using a hyperplane with the largest amount of margin. That is the reason why an SVM classifier is also known as a discriminative classifier. SVM finds an optimal hyperplane for helping to classify new data points. In general SVM is deemed to be a classification approach but it can be used in both classification and regression problems. It can easily handle multiple continuous and categorical variables. SVM constructs a hyperplane in multidimensional space to separate different classes. SVM generates an optimal hyperplane in an iterative manner, minimizing an error. The core idea of SVM is to find a maximum margin hyperplane (MMH), i.e., a hyperplane that optimally subdivides the dataset into classes (see for example Figure 4.3).

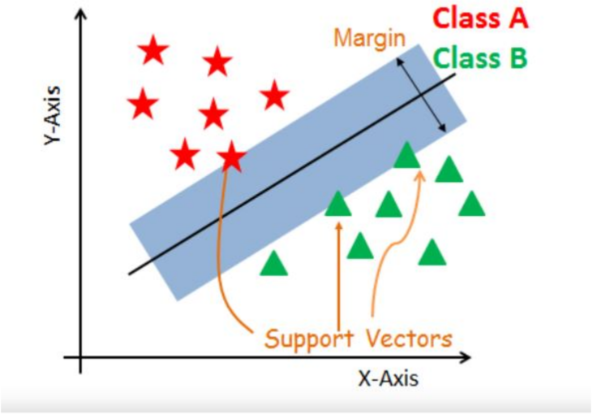
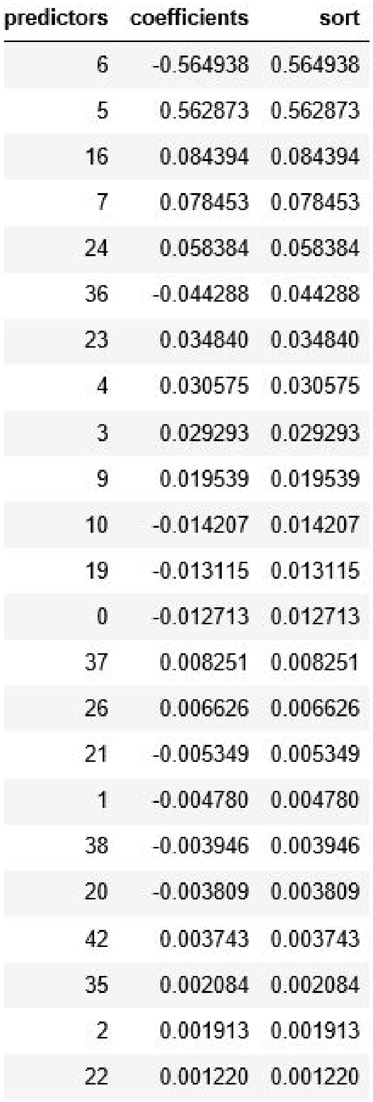


Figure 4.3: Example of SVM.

If one uses a **SVM** classifier with good hyper-parameters for important factors, one can obtain an accuracy as high as 81%. We also use the following metrics to select the best model: confusion matrix, precision, recall, and F1 scores.

Table 4.1



predictors	coefficients	sort
6	-0.564938	0.564938
5	0.562873	0.562873
16	0.084394	0.084394
7	0.078453	0.078453
24	0.058384	0.058384
36	-0.044288	0.044288
23	0.034840	0.034840
4	0.030575	0.030575
3	0.029293	0.029293
9	0.019539	0.019539
10	-0.014207	0.014207
19	-0.013115	0.013115
0	-0.012713	0.012713
37	0.008251	0.008251
26	0.006626	0.006626
21	-0.005349	0.005349
1	-0.004780	0.004780
38	-0.003946	0.003946
20	-0.003809	0.003809
42	0.003743	0.003743
35	0.002084	0.002084
2	0.001913	0.001913
22	0.001220	0.001220

Other classifiers

We tried other classifiers such as Gradient Boosting and Logistic Regression and we compared them with SVM and obtained the curves in Table 4.4.

In conclusion SVM offers a very high accuracy compared to other classifiers.

4.3.2 Survival models

Theory

In this subsection we use survival models to estimate the probability of a flight spill three days prior to the departure date. Indeed we can consider the duration of time t before the spill as the realization of a random variable T whose cumulative distribution function is F and density f. Since we have 356 days between the opening date of the flight and the departure date, we must have $T \leq 356$, meaning that the event “spill” does not necessarily happen before the date of departure (it is “right censoring”).

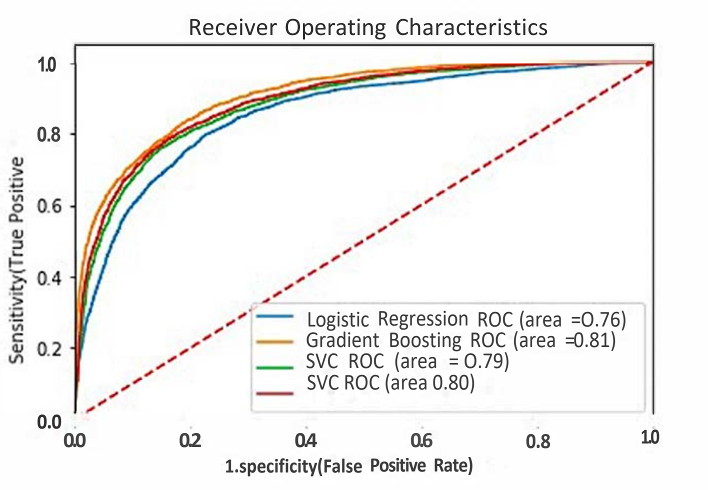


Figure 4.4: ROC.

We try to estimate the survival function at time t.

$$S(t) = \mathbb{P}(T > t) = 1 - F(t)$$

Survival models also seek to estimate the hazard function (denoted by h), i.e., the probability of occurrence of the event at time t. Here is the formula for h: $h(t) = \frac{f(t)}{S(t)}$. The most frequently used estimator of the survival function is the Kaplan-Meier estimator.

Since it seems logical for the spill probability of a flight to depend upon the departure time of the flight ("morning," "afternoon"), the day of the week, the week within the year, and several other variables, it is important to add explanatory variables. This requires the use of the Cox model. It should be noted that for using this model, we need to make assumptions in order to estimate the survival function. The most important of these is that the time dependence of the risk of observing the event is identical for all individuals. For the Cox model the hazard function can be written as follows, where h_0 is called the basic risk function and X is a set of supposed covariates.

$$h_i(t) = h_0(t)e^{X_i'\beta}$$

For reasons of time we have retained as covariates the variables mentioned above, but we strongly recommend including more relevant variables such as the number of bookings, 7 days before, that number 14 days before etc. To return to our problem, we wish to estimate the probability x days before the flight departure. Having already estimated the survival function, this probability is given by the following formula.

$$\mathbb{P}(T < 353 | T > x) = \frac{\mathbb{P}(x < T < 353)}{\mathbb{P}(T > x)} = \frac{S(x) - S(353)}{S(x)}$$

This probability can be computed if we know the estimator of S.

Applications

For the application part we considered a specific flight and held the value of x at 30 days. We remind the reader that we have many instances of a given flight during the year. For every instance we have information on the 356 days of reservations prior to departure. In our survival model for one flight, the individual i represents the flight instance. For each flight instance we have noted the time T_i representing the number of days before the spill. If a flight instance j never spills, we have $T_j = 356$. We also need a variable indicating whether a flight instance spills or not. We define the variable $\mathbf{Spill_i}$

as follows.

$$\mathbf{Spill_i} = \begin{cases} 1 \text{ if the flight experience } i \text{ spills} \\ 0 \text{ otherwise} \end{cases} .$$

Table 4.2 presents the list of variables that were used.

Table 4.2: Summary of variables used.

Variable label	Signification
T	Number of days before the spill
Spill	equals 1 if the flight instance spills and 0 otherwise
Moment_of_day	Moment of the day: “Morning” or “Afternoon”
Dow	Day of the week
Woy	Week of the year

Feature engineering: We first show how this method can generate variables that may be useful in machine learning models. Indeed the coefficients of the explanatory variables allow us to measure their effect on the probability of spill, all things being equal. This is achieved by computing e^{β_j} for the coefficient of covariate x_j. In the case of a dichotomous variable, for example, if $x = 1$ corresponds to an individual who receives a treatment and $x = 0$ otherwise, then e^{β} represents the relative risk over time of observing the event for an individual receiving the treatment in relation to an individual who does not receive it. The figure below shows the characteristics of 5 flights based on covariates.

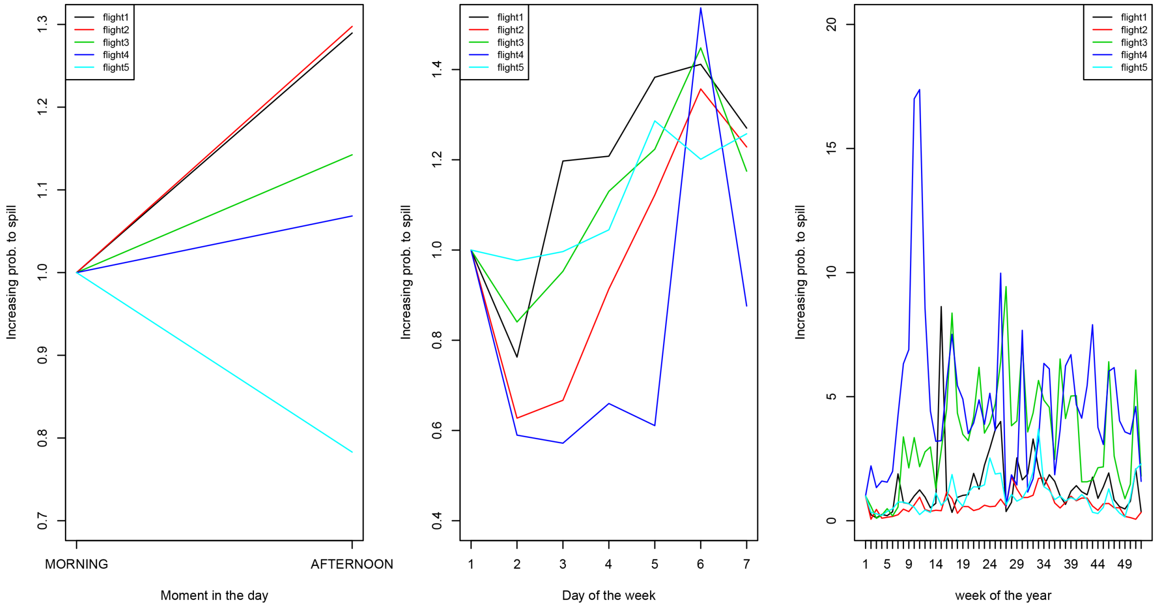


Figure 4.5: 5 flights characteristics.

We can see that flights 1,2,3, and 4 are more likely to spill in the afternoon while flight 5 is more likely to spill in the morning. Similar analyses can be performed for the Dow and Woy variables. This is the type of useful information characterizing the flight that can be used to improve another model.

Model performance: To evaluate the performance of the model, we used it on a training sample and checked its performance on a test sample. Application to one flight in order to predict the probability of spill 3 days before departure knowing that we are 30 days from departure yields a prediction score of 67.01% and a MSE of 53.17%.

Future work

As mentioned above, for an improvement of the predictive ability of the model, it is necessary to add relevant explanatory variables. In addition this method can help create useful variables in other models.

4.3.3 Kalman filtering approach

The Kalman filter is a powerful estimation and prediction method used in many technological fields (meteorology, radar, communications, etc.). A nice introduction to it can be found in [4].

First we use this method to predict the occupation rate at time $-D$ (i.e, the date occurring D days before the departure date). The spilling probability at time $-D$ can then be deduced in a straightforward fashion from the occupation rate.

Kalman model

In the following we consider the specific case of one-dimensional systems. The Kalman filter is derived from the Kalman model, which consists of two main components:

- a state equation constructed from a dynamical model and an associated process noise:

  $$\dot{x}(t) = ax(t) + bu(t) + w(t),$$

  where

  - x is the state of the system at time t,
  - $u(t)$ is the known deterministic input of the model,
  - $w(t)$ is the noise process;

- a measurement equation constructed from a model observation, linking measurements to the model states, associated with a measurement process noise:

  $$y(t) = cx(t) + du(t) + v(t),$$

  where

  - y is the measurement,
  - $v(t)$ is the measurement noise process.

In the case of linear dynamical systems, the Kalman filter is constructed from the discretized linear dynamical systems in the time domain: state x_{k+1} is derived from state x_k by applying a linear operator perturbed by errors.

The state sequence $\mathbf{x} := (x_k)_{k\geq 0}$ can be considered as a Markov chain built on a linear operator perturbed by errors. Thus the state equation associated with the Kalman filter is

$$x_{k+1} = a_k x_k + b_k u_k + w_k,$$

where

- a_k is the state transition model (allowing the transition from state x_{k-1} to state x_k);
- b_k is the control-input model, representing the time-varying external perturbations that are not taken into account by the model (e.g., the wind in the case of an airplane);
- w_k is the process noise, assumed to be drawn from a centred normal distribution $\mathcal{N}(0, q_k)$, where q_k is the variance of the process.

The measurement equation associated with the Kalman filter is

$$y_k = h_k x_k + v_k,$$

where

- h_k is the observation model, mapping the model state space into the observed space;
- v_k is the observation noise, assumed to be drawn from a centred normal distribution: $v_k \sim \mathcal{N}(0, r_k)$, where r_k is the variance of the observation noise.

The initial state and the noises at each step $\{x_0, w_1, \cdots, w_k, v_1, \cdots, v_k\}$ are all assumed to be mutually independent.

Once the discrete Kalman model have been written, we can apply its associated algorithm to make model predictions.

Kalman filter procedure

The Kalman filter is thus a recursive estimator in the sense that the full history of the observations and estimates is not needed. In what follows we introduce the notation $\hat{x}_{i|j}$ for denoting the state of the estimate x at time t_i given observations up to time $t_j \leq t_i$. The state of the filter is thus classically represented by two variables:

- the *a posteriori* state estimate at time t_k, $\hat{x}_{k|k}$, given observations up to time t_k, and
- the *a posteriori* error $\hat{p}_{k|k}$, corresponding to a measure of the estimated accuracy of the state estimate.

The procedure consists then in two main steps: the prediction step, where both the current state $\hat{x}_{k|k}$ at time k and the noise are predicted from the model, and the update step, where the Kalman filter is updated through the new measurement y_k at time k. Here is a summary of the computations.

1. Prediction step
   (a) Predicted (a priori) state estimate: $\hat{x}_{k|k-1} = a_k \hat{x}_{k-1|k-1} + b_k u_{k-1}$
   (b) Predicted (a priori) error covariance: $p_{k|k-1} = (a_k)^2 p_{k-1|k-1} + q_k$
2. Update step:
   (a) Innovation residual: $_k = y_k - h_k \hat{x}_{k|k-1}$
   (b) Innovation variance: $s_k = (h_k)^2 p_{k|k-1} + r_k$
   (c) Optimal Kalman gain: $g_k = p_{k|k-1} \frac{h_k}{s_k}$
   (d) Updated (a posteriori) state estimate: $\hat{x}_{k|k} = \hat{x}_{k|k-1} + g_k y_k$
   (e) Updated (a posteriori) estimate covariance: $p_{k|k} = (1 - g_k h_k) p_{k|k-1}$

Application to the occupation rate forecast prediction

The proposed Kalman filtering approach aims to infer the dynamics of the current booking rate from the knowledge of the past booking rate (1 year) and a readjustment of the trajectory based on daily measurements. In other words we model the occupation rate function and fit its associated parameter values using the observations made during year -1. Then we apply the Kalman filter procedure to update the prediction with the current partial observations: e.g., we have measurements up to time -60 and we want to predict the occupation rate three days before departure).

We recall that our observations consist of longitudinal measurements $x := (x_k)_{k=1,\cdots,K}$, where $x_k := f(t_k)$ and $k \in \mathbb{N}^*$. The time sequence $t = (t_k)_{k=1,\cdots,K}$ is not necessarily homogeneous (i.e.,

$t_{k-1} - t_k$ is not fixed for all k). Here the occupancy rate is a composite variable defined as the ratio between the number of bookings and the airplane capacity.

The first step of the Kalman filtering consists of building a model to represent the behaviour of the data. To that end we consider several models.

Model 1: polynomial curve In our case the occupation rate forecast is a curve whose ordinate first equals 0 (X days before departure) then reaches 1 (and sometimes a little more). Hence a logistic curve is a natural choice and we first consider a Gompertz curve to represent the occupation rate forecast dynamics. It appears that such a model cannot be fitted with the solvers proposed by the languages Python and R (because of a non-invertible hessian matrix).

To mimic the logistic curve we consider a higher-degree-of-freedom model, i.e., a polynomial function, specifically a polynomial of degree five:

$$P(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5.$$

We estimate the coefficients a_i, $i = 1, \cdots, 5$ using the Python solver *polyfit* in the package *numpy*. To derive the state transition function a_k for each time t_k from our nonlinear polynomial function, we apply an Euler explicit scheme. In other words we linearize our model and obtain:

$$\frac{x_{k+1} - x_k}{t_{k+1} - t_k} = \dot{P}(t_k) \Rightarrow x_{k+1} = (t_{k+1} - t_k)\dot{P}(t_k) + x_k = a_k x_k + \gamma_k.$$

Note that this is not a linear operator but an affine one!

We apply the algorithm proposed in subsection 4.3.3 with the following values: $h_k = 1$, $q_k = 0.3$, and $r_k = 0.3$. Usually these parameters can be fitted with an EM approach. Since we always consider large time windows (often 170 days), we do not need an optimal fit at the beginning.

Model 2: speed approximation Modelling the occupation rate can also be carried out by representing directly the speed dynamics a_k. To that end we propose to model the speed dynamics through a highly stable Euler numerical scheme built with a sliding window. In other words the states x_k verify the following equation.

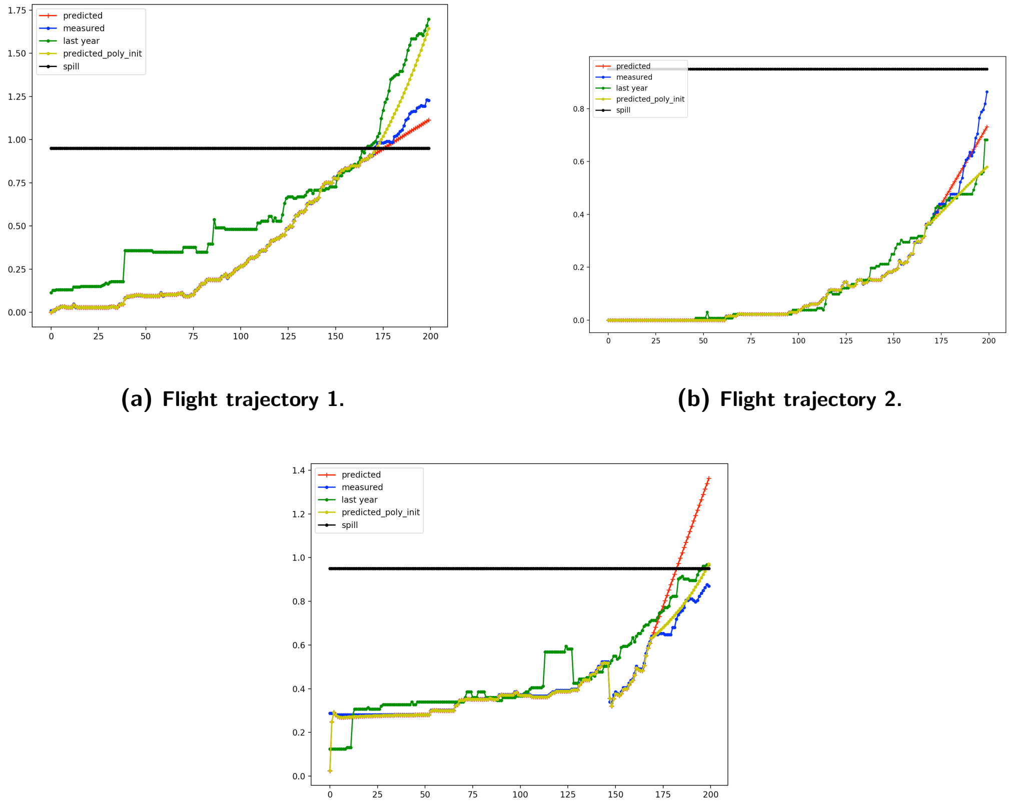
$$\frac{(x_{k-2} - 8 * x_{k-1} + 8 * x_{k+1} - x_{k+2})}{12} = \dot{P}(t_k).$$

Note that this time discretization is more stable than the Euler explicit scheme used to linearize Model 1.

Results Figure 4.6 displays some examples of the occupation rate forecast for the thirty last days (in year 0), based on a dynamical model (either Model 1 or Model 2) fitted on the -1 year measurements and updated through the Kalman filtering approach with data up to -30 days before departure. We observe that the forecast seems to follow the measurements (i.e., the red, yellow, and blue lines are qualitatively "close" and have the same shape). In some cases, however, this qualitative similarity is not sufficient to predict the spilling three days before departure. For instance Model 2 (red line) predicts a spill almost fifteen days before departure while this is not observed on the 0-year data (false positive answer). A false positive answer has also been observed with Model 1 (see details in the next paragraph).

Prediction of the spilling probability We deduce the spilling probability from our model predictions. We consider that a flight spills three days before departure with a threshold probability of at least 0.95 if the predicted occupation rate on day -3 is greater than 0.95.

Figure 4.7 summarizes the accuracy of our method by comparing the proportion of spilling cases predicted by Model 1 (resp. Model 2) with the true spilling cases proportion. It appears that the two approaches are "good" (the prediction score is more than 75% of correct answers). In particular the amount of false negatives (meaning that the model does not predict a spill while actually there is one) is around 12% for Model 1 and 19% for Model 2.



(a) Flight trajectory 1.

(b) Flight trajectory 2.

(c) Flight trajectory 3.

Figure 4.6: Results obtained with the Kalman filtering approach for three flight trajectories with Models 1 and 2. For each flight trajectory we apply the Kalman filtering method presented in 4.3.3 with both Model 1 (yellow lines) and Model 2 (red lines). The dynamical model (i.e., Models 1 and 2) is fitted on the year -1 and we update the 0-year trajectory using measurements up to 30 days before departure to forecast the occupation rate for the last thirty days before departure. The blue lines correspond to the current measurements (year 0) while the green lines are the measurements carried out in year -1 and on which Models 1 and 2 are fitted. The spilling threshold is represented by the black lines.

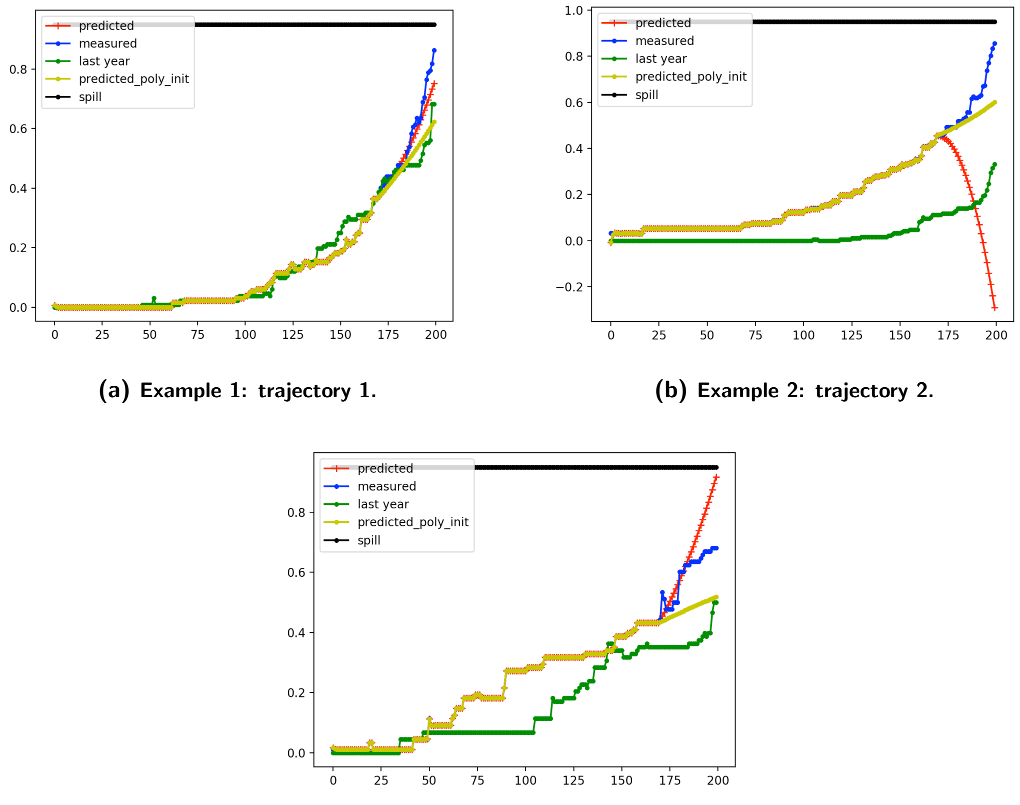
	Actual	Predicted Model 1	Predicted Model 2
Spill occupation rate	37%	39%	27%
Prediction score		71%	72%
False negative		12%	19%

Figure 4.7: Spilling probability prediction for the "three days before departure" date (threshold: 0.95%). For each flight trajectory within a dataset composed of 11,307 flight trajectories, we compute whether or not the flight spills three days before departure. We consider that a flight spills with probability at least 0.95% if its occupation rate forecast three days before departure is greater than 0.95.

Going further: estimating the model parameters

The Kalman filtering approach can be used not only to adjust the trajectory of a dynamical model by taking the measurements into account, but also to obtain better estimates of the model parameters. We consider this last option in the following. Instead of estimating the states $x = (x_k)_{k \geq 0}$ of the occupation rate trajectory, we apply the Kalman filtering approach to fit the polynomial parameters a_i (for $i = 1, 2, 3$) directly.

Figure 4.8 illustrates the results obtained with this method (red lines). The results obtained are mixed: we have a better occupation rate prediction (the blue and red lines are closer) with this model for the flight trajectories 1 and 2, while the flight trajectory 3 prediction is not accurate (the model predicts a spill while it is not the case with the true measurements). This phenomenon may be due to the high sensitivity of the parameters: a small change in the polynomial coefficients may lead to highly different dynamics.



(a) Example 1: trajectory 1.

(b) Example 2: trajectory 2.

(c) Example 3: trajectory 3.

Figure 4.8: Results obtained with the Kalman filtering approach. We test our Kalman filtering approach with three trajectories randomly selected from the dataset. The blue, green, yellow, and dark lines are defined in the same fashion as in Figure 4.6. The red lines are obtained by fitting the polynomial coefficients (a_i for $i = 1, 2, 3$) through the Kalman filtering algorithm.

Kalman filtering approach: conclusion and perspectives

In this subsection we have proposed several approaches based on Kalman filtering to predict the occupation rate and derive the spilling probability. This approach appears to be promising but the accuracy of the dynamical model representing the occupation rate needs to be enhanced, since it is the key to the success of such an approach. For instance the approach consisting of updating the model parameters through Kalman filtering requires a model with a low parameter sensitivity.

Our approach predicts the occupation rate of a flight trajectory by using past information, since the dynamical model is built from the "year -1" trajectory. A further step would be to predict the occupation rate in the case of a new flight trajectory that shares some characteristics with known flights. In other words we aim to construct a dynamical model of a given flight trajectory *a posteriori*: this model would be based on closed trajectories and use for instance machine learning techniques (random forests) to compute the model parameters. This approach requires a model with low parameter sensitivity.

4.4 Conclusion

During the IPSW we proposed several complementary approaches to predict the flight spill. Our machine learning approaches allow us to predict the spill of a flight three days before departure (random trees and SVM) and select the appropriate features (Lasso). Our two modelling approaches (survival analysis and Kalman filtering) provide us with encouraging prediction rates (greater than 67 %). The next step will be to use the results from the machine learning approach to add new features in our models.

Bibliography

[1] M. Berge and C. A. Hopperstad. Demand driven dispatch: A method for dynamic aircraft capacity assignment, models and algorithms. Operations Research, 41, 1993.

[2] SIAM Journal on Scientific and Statistical Computing, editors. Linear inversion of band-limited reflection seismograms. Siam edition, 1986.

[3] Robert Tibshirani. Regression shrinkage and selection via the lasso. 58 (1) : 267–88, 1986.

[4] Felix Govaers, editor. Introduction and Implementations of the Kalman Filter. Intechopen edition, 2019.

[5] John P Klein and Melvin L Moeschberger. Survival analysis: techniques for censored and truncated data. Springer Science & Business Media, 2006.

[6] Simon Quantin. Modèles semi-paramétriques de survie en temps continu. 2019.

5 Optimizing the design of a loyalty program

Federico Bobbio [a]
Margarida Carvalho [a]
Adel Nabli [a]
Sriram Sankaranarayanan [b]
Simon Germain [c]
Mohamed Ossama Hassan [c]
Bassirou Ndao [c]
Geneviève Pagé [c]
Jeremy Piche-Bisson [c]
Vincent Purenne [c]
Russel Shaul [c]

[a] *Université de Montréal, Montréal (Québec), Canada*

[b] *GERAD & Polytechnique Montréal, Montréal (Québec), Canada*

[c] *Aeroplan, Montréal (Québec), Canada*

October 2020
Les Cahiers du GERAD

5.1 Introduction

Overview. Aeroplan is one of the biggest loyalty programs in Canada with more than five million members. In exchange for their participation in the program the members are offered miles each time they make purchases from the Aeroplan partners. Those miles can then be used by the members as an alternative currency allowing them to buy a variety of products and services offered by the pool of partners. In this pool Air Canada holds a unique place, since it has acquired Aeroplan in January 2019: it is thus both a partner and the owner of Aeoplan.

The purpose of our work within the *The Ninth Montreal Industrial Problem Solving Workshop* was to develop a methodology for optimizing the design elements of the loyalty program, i.e., maximizing profitability of the program and its attractiveness and value for its members over the long term. Our team's goal was to build a conceptual model that would integrate key program dynamics in a game theory framework.

Organization and contributions. This report is organized in the following way: first, in Section 5.2, we will broadly describe how the members, Aeroplan, and the partners interact and also describe the cash-flows from each agent's point of view. Then in Section 5.3 we will give more details to define rigorously the agents (players), the actions they can take, and the corresponding variables they can manipulate to define their respective strategies. Afterwards, using all the defined elements, we will formulate the different objectives of the players along with the constraints on their actions. Finally, in Section 5.4, we will specify the game the agents are playing and give an outline of how to solve the coupled optimization problems: this is expected to give crucial insights into the most rational strategies to be followed by the players. Section 5.5 draws conclusions and point to future research directions.

This report represents a further step in the decision-making support and guidance that Aeroplan needs when negotiating with partners, as well as designing its loyalty program.

5.2 Preliminaries: agents interactions

In what follows we describe each agent and the interactions between agents.

5.2.1 The members

When enrolled in the fidelity program a member has the opportunity to accumulate miles when buying a product from a partner. This *accumulation* of miles by the members can then be used as an alternative currency to buy products from the partners (this action is called a *redemption*). From the members' point of view these are the only two actions possible, but what complexifies the understanding of loyalty programs is their dynamics over time. Indeed different members could behave quite differently: some may prefer to make frequent purchases to increase their accumulated miles, waiting a long time for the ability to redeem a valuable good, while others may redeem their miles as they accumulate them. The different types of *"accumulation - redemption"* cycles that may occur imply that a realistic mathematical model must take the time variable into account and separate the members into different behavioural segments.

5.2.2 Aeroplan's cash-flow

We can think of Aeroplan as a bank that manipulates two different currencies: miles and dollars. Aeroplan's income arises from the sale of miles to partners. Each time a member spends dollars on a purchase from a partner, the partner grants this member some miles; the precise amount is fixed by the partner. In order to be able to give those miles away the partner first needs to buy them from Aeroplan, at an exchange rate fixed by contract. On the other hand Aeroplan's members can spend the miles they have gathered to buy products from the partners, at a price in miles fixed by Aeroplan. For the members to actually receive the good they want, however, Aeroplan must purchase the said good from the partner. Hence Aeroplan's profit comes only from the difference between the amount of dollars it manages to generate from the sale of miles to partners and the amount of dollars it has to spend on the purchasing of products for the members. In addition to those flows Aeroplan spends some money on advertising the partners' products to its members, both through spending dollars on marketing and offering "free" miles for short period of times on some products. Indeed, the more the members buy products, the more miles the partners will have to distribute (and thus purchase from Aeroplan first). Finally, since offering a large pool of products to buy with miles is something that appeals to the members, Aeroplan also advertises its services to gain new partners. We sum up those considerations in the following formula. Note that "AE" stands for 'Aeroplan."

$$\begin{aligned}\text{profit}_{AE} &= \text{income}_{AE} - \text{costs}_{AE}\\ \text{profit}_{AE} &= \text{income from selling miles to partners}\\ &\quad - \text{cost of buying products from partners for the members}\\ &\quad - \text{cost of advertisement}\end{aligned}$$

5.2.3 The partners' cash-flow

The partners hope that being in the loyalty program will increase their sales both directly (members will prefer to buy from them because it allows them to accumulate miles) and indirectly (members will buy their products through redemption). This increase in sales will entail a profit increase for the partners. There are two kinds of costs for the partners. First they have to buy miles from Aeroplan to be able to give them to the members, and secondly they have to advertise their products to the program members to incite them to purchase these products. Hence a partner's profit can be expressed as follows.

$$\begin{aligned}\text{profit}_{p} &= \text{income}_{p} - \text{costs}_{p}\\ \text{profit}_{p} &= \text{direct profit from additional clients}\\ &\quad + \text{profit from selling products to AE for redemption}\\ &\quad - \text{cost of buying miles from AE}\\ &\quad - \text{cost of advertisement}\end{aligned}$$

5.3 The model

In order to propose a mathematical model of the dynamics between agents, we will enumerate all the agents, detail the actions they can take, and define rigorously their objectives and constraints.

5.3.1 Agents

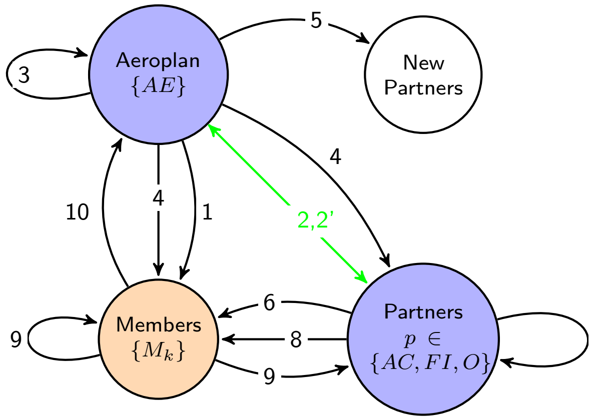
In our model, we will consider five subgroups of agents: Aeroplan (AE), the members ($\{M_k\}$, where each M_k represents a segment of members that behave in the same manner), the partner and owner Air Canada (AC), the partners that are Financial Institutions (FI), and the other partners (O). The reason of this distinction between partners lies in the fact that Air Canada is a special partner as it is also the owner, and the Financial Institutions don't sell any product that can be obtained by spending miles (contrary to the others partners): their only revenue comes from the use of their credit cards by the members. In the end we can express the set of agents as

$$I = \big\{AE,\ \{M_k\},\ AC,\ FI,\ O\big\}.$$

Thus we can identify three groups of agents: Aeroplan, the members, and the partners. To refer to an agent we will use the subscript p with $p \in \{AC, FI, O\}$.

5.3.2 Actions

The set of possible actions an agent can take depends on the group to which it belongs. In Figure [5.1] we summarize the different actions each agent can take. The details of these interactions are discussed in the following sections.



Legend:

🔵 : Agents for which the strategy is determined by solving an optimization problem

🟠 : Agents for which the strategy is determined by a regression model

↔ : Actions fixed by an agreement between the agents *(corresponding variables fixed in the optimization problem)*

→ : Actions determined by the left agent that directly affect the right agent

Id	Variable's meaning	Var	Domain	Unit
1	Nb of miles AE requires for a member to redeem product j	m_j^{red}	$\in \mathbb{R}$	miles
3	Total budget AE allocates to advertisement	B_{AE}	$\in \mathbb{N}$	\$
4	Fraction of B_{AE} used to advertise product j of partner p	$a_{p,j}$	$\in \mathbb{N}$	\$
5	Fraction of B_{AE} used to advertise to get new partners	Ψ	$\in \mathbb{N}$	\$
6	Nb of miles a member accumulates by buying product j	m_j^{acc}	$\in \mathbb{N}$	miles
7	Total budget partner p allocates to advertisement	B_p	$\in \mathbb{N}$	\$
8	Fraction of B_p used to advertise product j of partner p	$b_{p,j}$	$\in \mathbb{N}$	\$
9	Nb of miles members M_k accumulated by buying product j from partner p	$d^{acc}_{M_k,p,j}$	$\in \mathbb{N}$	miles
10	Nb of miles members M_k redeemed by buying product j from AE	$d^{red}_{M_k,p,j}$	$\in \mathbb{N}$	miles

Id	Parameter's meaning	Param	Domain	Unit
2	Amount of dollars that product j costs to AE	$\overset{AE\to P}{\pi_j}$	$\in \mathbb{R}$	\$
2'	Price (in dollars) of buying 1 mile from AE for partner p	$\overset{P\to AE}{\pi_p}$	$\in \mathbb{R}$	\$/miles

Figure 5.1: Graph representing the agents, the actions they can take, and the agents affected directly by their actions (simplified representation for the sake of clarity: in reality all actions affect everyone indirectly to some extent).

Aeroplan's actions

There are four types of actions for Aeroplan. The most critical one is setting, for each product j in the pool of products J sold by the partners, a conversion rate $m_j^{red} \in \mathbb{R}$ of miles to product: for each product a member can buy using miles (a *redemption*), Aeroplan sets the number of miles needed to buy the product (this product costs a certain amount of dollars $\overset{AE\to P}{\pi_j}$, fixed by contract with the partners).

Aeroplan must also decide how many dollars to allocate to its advertisement budget, denoted by $B_{AE} \in \mathbb{N}$. What should be the value of $a_{p,j}$, the fraction of the budget allocated to advertise product j sold by partner p? What should be the value of Ψ, the share of the budget spent on trying to find new partners ?

Thus the action space for Aeroplan is

$$X_{AE} = \mathbb{R}^{|J|} \times \mathbb{N} \times \prod_{p \in \{AC,FI,O\}} \mathbb{N}^{|J_p|} \times \mathbb{N}.$$

Partners' actions

There are three types of actions that a partner can take. First each partner p, for each product j in the set J_p of products it is offering, has to set the number $m^{acc}_{p,j} \in \mathbb{N}$ of miles it will grant a member when he (she) buys this product in dollars. The partner p also decides what amount to allocate to advertising ($B_p \in \mathbb{N}$) and the fraction of this budget spent on advertising for each of the products it sells (denoted by $b_{p,j} \in \mathbb{N}$ for product j). The variable $b_{p,j}$ accounts for the efforts in both marketing and promotion.

Remark 1 *During a promotional event products sold by the partners grant a greater amount of miles than usually. The extra miles given to members are a shared effort between Aeroplan and the Partners, included in the variables $a_{p,j}$ and $b_{p,j}$. To represent the "promoted" version of the product (which grants an amount of miles different from the standard product), we add a new product j to the pool J_p with its associated $m^{acc}_{p,j}$.*

The action space for the partner p is

$$X_p = \mathbb{N}^{|J_p|} \times \mathbb{N} \times \mathbb{N}^{|J_p|}.$$

Members' actions

The members only have two choices: purchasing new products that will give them miles or burning the miles they have already accumulated in order to purchase new products through Aeroplan's platform. To represent the miles that a segment of members M_k accumulates through buying the product j from partner p, we will use the variable $d^{acc}_{M_k,p,j} \in \mathbb{N}$, and to represent the amount of miles it redeems for a product j from partner p, we will use the variable $d^{red}_{M_k,p,j} \in \mathbb{N}$. Since not all products are redeemable, we denote by J^{red}_p the set of products from partner p that are eligible for redemption. Hence the action space for the member segment M_k is

$$X_{M_k} = \prod_{p \in \{AC,FI,O\}} \mathbb{N}^{|J_p|} \times \prod_{p \in \{AC,FI,O\}} \mathbb{N}^{|J^{red}_p|}.$$

5.3.3 Objectives and constraints

Aeroplan

Aeroplan has several objectives: as a standalone company it tries to make profit, but as a property of Air Canada it tries to make AC profitable. First we focus on Aeroplan. We can model the objective of AE with several variables. Aeroplan's income arises from the miles it manages to sell to its partners. The rate of conversion of miles to dollars $\overset{P\to AE}{\pi_p}$ is fixed by contract with each partner p. The fraction of AE's income due to partner p can be written as the product of *"the number of miles this partner has bought"* and *"the rate of conversion of miles to dollars for p"*. The total income is then the sum of the incomes generated by all the partners.

$$\text{income}^{acc}_{AE} = \sum_k \sum_{p \in \{AC,FI,O\}} \sum_{j \in J_p} d^{acc}_{M_k,p,j} \overset{P\to AE}{\pi_{p,j}} \tag{5.1}$$

The expenses of AE are of several types. First it needs to buy products from partners for the members who wish to redeem their miles. Each product j a partner p has available for redeeming has a cost $\overset{AE\to P}{\pi_{p,j}}$ in dollars that is fixed by agreement between the partner and Aeroplan. The number of miles $m^{red}_{p,j}$ requested from a member who wishes to redeem a given product j is set by Aeroplan only. Then the cost for Aeroplan of exchanging members' miles for products is given by the following expression.

$$\text{cost}^{red}_{AE} = \sum_k \sum_{p\in\{AC,FI,O\}} \sum_{j\in J^{red}_p} d^{red}_{M_k,p,j} \frac{\overset{AE\to P}{\pi_{p,j}}}{m^{red}_{p,j}}. \tag{5.2}$$

To this expense we can add the one coming from advertising for the products (represented by $a_{p,j}$) and the one coming from advertising to new partners (Ψ), which we gather in the budget B_{AE}.

$$\text{cost}^{advertisement}_{AE} = B_{AE}. \tag{5.3}$$

On the other hand, since having many partners is something deemed attractive to the members and sought by Aeroplan, we can also take into account an income $f(\Psi)$ that is a consequence of the effort put into having diverse partners. The precise form of this income requires further research and thus for now it is modelled with a generic function f.

We model the short-term profit by summing the incomes and costs listed above. To enforce a "longer-term view" we can add to the Aeroplan's objective the fact that it wants members to be actually engaged for a long period of time, i.e., it wants to build members' loyalty. We represent this objective as a function of the accumulated miles denoted by $g(d^{acc}_{p,j})$.

Overall Aeroplan's objective is a trade-off between gaining instant profit and building its members' loyalty. This trade-off is represented by the parameter $\lambda \in [0,1]$ in the objective (5.4). To represent the influence of Air Canada in the objective of Aeroplan, we remove the *"profit made on the back of Air Canada"* from the objective of AE and add to this objective the profit made by selling plane tickets to members. (This amounts to adding to Aeroplan's objective the objective function of its "partner" AC: see the objective function (5a) in the partner's model). What remains for the Air Canada part in the objective is the cost of redeeming plane tickets and the profit generated from having members accumulate miles by buying tickets. There is a production cost to exchanging a plane ticket for miles, so that each time a member redeems a plane ticket the union $AE \cup AC$ loses money. The generation of profits is expected to arise because members prefer to buy plane tickets from AC to accumulate miles and the amount of these sales exceeds the losses entailed by redemption.

$$max_{m^{red}_{p,j},B_{AE},a_{p,j},\Psi}\lambda \left(\sum_k \left(\sum_{p\in\{FI,O\}} \sum_{j\in J_p} d^{acc}_{M_k,p,j} \overset{P\to AE}{\pi_{p,j}} - \sum_{p\in\{AC,FI,O\}} \sum_{j\in J^{red}_p} d^{red}_{M_k,p,j} \frac{\overset{AE\to P}{\pi_{p,j}}}{m^{red}_{p,j}}\right)\right. \tag{5.4a}$$

$$\left. - B_{AE} + f(\Psi) + \sum_k \sum_{j\in J_{AC}} (p^{acc}_{j,k} s^{acc}_{j,k} + p^{red}_{j,k} s^{red}_{j,k}) - B_{AC}\right) \tag{5.4b}$$

$$+ (1-\lambda)\left(\sum_k \sum_{p\in\{AC,FI,O\}} \sum_{j\in J_p} g(d^{acc}_{M_k,p,j})\right) \tag{5.4c}$$

$$\tag{5.4d}$$

subject to

$$\sum_{p\in\{AC,FI,O\}} \sum_{j\in J_p} a_{p,j} + \Psi \quad \leq \quad B_{AE} \tag{5.4e}$$

$$\sum_k \sum_{p\in\{AC,FI,O\}} \sum_{j\in J_p} d^{red}_{M_k,p,j} \quad \leq \quad \sum_k \sum_{p\in\{AC,FI,O\}} \sum_{j\in J_p} d^{acc}_{M_k,p,j} \tag{5.4f}$$

$$a_{p,j} \quad \geq \quad l_{p,j} \quad \forall p \in \{AC, FI, O\}, j \in J_p \tag{5.4g}$$

The constraints listed above guarantee, respectively, that:

- the total advertising cost (advertising of products and advertising to partners) does not exceed the available budget;
- the number of miles redeemed is at most the number of miles accumulated within a given time period; and
- the budget spent advertising a given product is at least equal to some lower bound.

Partners

The objective of each partner p consists of three elements. The first element is the opposite of the profit made by Aeroplan, i.e., the profit of Aeroplan is a cost for the partner. Indeed the partner is hoping that even though the partnership with Aeroplan entails costs, this expense is less than the profit generated by the increase in sales due to the loyalty program, which incites the members to consume. The second element of the objective is the profit generated by the sale of its products to the members of the loyalty program. We make a distinction between the profit p_j^{acc} generated by the purchase of product j and the profit p_j^{red} generated through redeeming of miles. To compute the total profit generated by sales we multiply the unitary profit $p_{j,k}$ of product j by the total number $s_{j,k}$ of units of product j purchased by member k. (We make the distinction between the amount $s_{j,k}^{acc}$ purchased with dollars and the amount $s_{j,k}^{red}$ obtained through redeeming). We then take the sum over all products and clients. Finally the third element of the objective is the total budget B_p allocated to advertising by the partner. We obtain the following model.

$$\max_{m_{p,j}^{acc}, B_p, b_{p,j}} \sum_k \left(\sum_{j \in J_p} \left(p_{j,k}^{acc} s_{j,k}^{acc} - d_{M_k,p,j}^{acc} \overset{P \to AE}{\pi}{}_{p,j} \right) + \sum_{j \in J_p^{red}} p_{j,k}^{red} s_{j,k}^{red} \right) - B_p \tag{5.5a}$$

subject to

$$\sum_{j \in J_p} b_{p,j} \leq B_p \tag{5.5b}$$

$$d_{M_k,p,j}^{acc} = s_{j,k}^{acc} m_{p,j}^{acc} \qquad \forall k, \forall j \in J_p \tag{5.5c}$$

$$d_{M_k,p,j}^{red} = s_{j,k}^{red} m_{p,j}^{red} \qquad \forall k, \forall j \in J_p \tag{5.5d}$$

$$\sum_k s_{j,k}^{acc} + s_{j,k}^{red} \leq \alpha_j \qquad \forall j \in J_p \tag{5.5e}$$

$$\sum_k s_{j,k}^{red} \leq \beta_j \qquad \forall j \in J_p \tag{5.5f}$$

The constraints listed above guarantee, respectively, that:

- the sum of the costs of advertising the products is less than or equal to the budget available;
- the number of miles a member k accumulates by purchasing item j from partner p is equal to the purchased quantity of this item times the number of miles attached to this product. The same holds for the redeemed products;
- the total purchased quantity of a product j is less than or equal to the available stock α_j; and
- the partner has a limited stock capacity (β_j) for those instances of product j that can be acquired through redeeming miles.

Members' behaviour

We decided to use a regression model to capture the complex behaviour of the members of the loyalty program. Indeed it is difficult to know exactly what members are optimizing when they make a decision.

Moreover we can consider that the behaviour of the members is only a reaction to the decisions taken by the leaders (i.e., Aeroplan and its partners). Hence the goods purchased by a particular segment M_k of members (and the quantities they buy with dollars and miles) are consequences of the decisions made by Aeroplan and its partners (and reflected in the values of model variables). To ascertain how the members react to different strategies, we will use some historical data on the loyalty program to fit a regression model for each segment of members. Next we provide guidelines on the variables influencing the two actions the members may choose (i.e., buying and redeeming).

First we consider $s^{acc}_{k,p,j,t}$, the amount of product j the segment of members M_k purchases in dollars from partner p at time t. This variable is a function of $m^{acc}_{p,j}$, the number of miles accumulated by buying product j, the budgets $a_{p,j}$, $b_{p,j}$ spent by Aeroplan and the partner (respectively) to advertise and promote the product, and a measure of how interesting it is to accumulate miles at that point in time. This measure depends on what those miles could allow the member to obtain, i.e., the set $\{m^{red}_{p,j,t-1}\}_{p,j}$ of prices in miles of all the products in the previous period. Hence we have the following relationship.

$$s^{acc}_{k,p,j,t} = \mathcal{R}(\{m^{red}_{p,j,t-1}\}_{p,j}, m^{acc}_{p,j,t}, a_{p,j}, b_{p,j}) \tag{5.6}$$

Second we consider the redemption of their miles by the members. The quantity of the product j that will be bought with miles at time t is denoted by the variable $s^{red}_{k,p,j,t}$. This variable depends on the number of miles accumulated previously $d^{acc}_{k,p,j,t-1} = s^{acc}_{k,p,j,t-1} m^{acc}_{p,j,t-1}$, the current prices in miles of the product $\{m^{red}_{p,j,t}\}_{p,j}$, and on the efforts $a_{p,j}$, $b_{p,j}$ put into advertising the given product. This leads to the following relationship.

$$s^{red}_{k,p,j,t} = \mathcal{R}(\{m^{red}_{p,j,t}\}_{p,j}, s^{acc}_{k,p,j,t-1}, m^{acc}_{p,j,t-1}, a_{p,j}, b_{p,j}) \tag{5.7}$$

Remark 2 *Some other parameters could be useful here in order to predict the members' behaviour (e.g. the income level, the literacy in the program, to name a few).*

5.4 The game between the agents

We now focus on finding optimal strategies for Aeroplan and its partners. Let us first describe two major obstacles.

- The optimization problems of the previous section are most likely **nonlinear** and **non-convex**. Indeed the values s^{red} and s^{acc} appear in each problem. These values are actually given by regression models, which can be very complex, making the optimization problems tricky to handle.
- The optimization problems are **coupled**. Indeed some of the variables over which a given agent optimizes also appear in the optimization problems of the other agents. This is the case for the variables $m^{red}_{p,j}$ and $m^{acc}_{p,j}$, which appear in every objective. Thus the problems cannot be solved independently: they must be solved simultaneously.

To overcome the first obstacle we can borrow some ideas from [1], where the authors describe a way to transform the non-linearity appearing in our problem in a manageable way, provided the regression model is a neural network using only ReLU and max-pooling non-linearities.

In the following paragraphs we focus on the second obstacle. Coupled optimization problems between agents can be interpreted as a game played between the agents. Thus we could use some results from game theory to help us tackle the issue of finding optimal strategies. In order to do that we first need to clarify what we are looking for, i.e., formalize what constitutes an *optimal strategy*. Since several definitions are reasonable, we will try to explore some of them and give some indications on how to find optimal strategies in each setting.

First let us recall the definition of strategy. Let I be a set of N players (agents) and i a given player: x_i is called a **pure strategy** if it assigns values to the decision variables controlled by player i

and x_i is *feasible* (i.e., this assignment respects the constraints of the model). For instance x_i could be a decision vector feasible for one of the mathematical programming formulations in Section 3.2. A **mixed strategy** would be a probability distribution on this feasible action space.

In Section 5.4.1 we will assume a non-cooperative setting, i.e., the players are assumed to be looking for their own interest only. An alternative point of view is to consider cooperation, where all of them together search for a *joint strategy* that will maximize their payoff for the whole group *(or for coalitions of players)*: this joint strategy could yield a better overall payoff. The challenge of the cooperative approach is that it does not specify the individual payoff of every player, so it requires a policy to allocate the surplus arising from the coalition of players. This point of view will be the focus of Section 5.4.2.

5.4.1 Non-cooperative model

In this model an "optimal strategy" is actually an *equilibrium*, or as defined in [2], a profile of strategies where no agent has an incentive to change his (her) behaviour. We will focus on *Nash equilibria* and *correlated equilibria*. A Nash equilibrium is a *strategy profile*, i.e., an assignment of strategies, one for each agent, such that no individual player has an incentive to deviate to another strategy: in other words, there is no gain in terms of payoff in deviating from the current strategy. Therefore a Nash equilibrium is a strategy profile such that every agent's strategy is his (her) best reaction to the $N-1$ opponents' strategies.

Remark 3 *In non-cooperative game theory an equilibrium might be given by mixed strategies, that is, a probability distribution on each player's set of feasible strategies.*

The notion of correlated equilibrium extends the notion of Nash equilibrium and has some practical and computational advantages over Nash equilibria. Indeed all the Nash equilibria are correlated equilibria but for a given game, there may exist strategy profiles in the set of correlated equilibria that are not Nash equilibria. Extending the notion of Nash equilibrium can be beneficial because the set of Nash equilibria sometimes only contains strategy profiles that are non desirable in practice. We clarify this through an example before diving into the definition of correlated equilibrium.

Example 1 *(The traffic game) Let us suppose we have a crossroad and two drivers arrive at the crossroad(drivers A and B), one on each road. Each of the players has only two pure strategies: either pass or wait. If both pass they crash into each other; if both wait nobody passes. The only solution profiles that "make sense in practice" are either A passes and B waits or B passes and A waits. Usually in those games both solution profiles are Nash equilibria, but so is the mixed strategy where A passes with probability* $1/2$ *and so does B. In this setting the solution profile "both pass at the same time" has a non-zero probability, which leads to a crash. In summary the set of Nash equilibria only contains the solutions A always passes and B always wait, at all times (which would block one road), the reverse, and a solution profile of mixed strategies that makes the possibility of a crash possible. None of those three solutions is good in practice. What we want is a solution that switches between the two pure Nash equilibrium strategies, i.e., sometimes A passes and B waits, sometimes the reverse holds. This equilibrium is usually not present in the set of Nash equilibria but is contained in the set of correlated equilibria.*

Remark 4 *For the sake of simplicity we did not define the payoffs rigorously in this example: a more detailed description of this example can be found in* ***Section 2*** *of [7].*

To define the notion of correlated equilibrium we switch from each player's individual point of view to a more global one. Here we consider that each player has only a finite set of pure strategies, and instead of players being restricted to strategies that are independent probability distributions on their respective sets of actions (mixed strategies), we consider any distribution on the set of all pure strategy profiles $\mathcal{S} = \prod_{i=1}^{N} \mathcal{S}_i$. Then a correlated equilibrium is a distribution σ^* on $\mathcal{S}$ such that, for each agent i, for each pure strategy x_i in $\mathcal{S}_i$ *(the set of pure strategies for a_i that appear in the support of σ^*)*, there is no incentive for agent i to deviate from the pure strategy x_i given that every other agent acts according to σ^* conditioned on x_i, i.e., acts as if he (she) believed agent i followed x_i.

Remark 5 *In practice we need to be more careful and technical in defining rigorously the notion of correlated equilibrium. The complete formal setup (including some σ-field notions) can be found in the original paper [3].*

Remark 6 *We can see that the definition of correlated equilibrium includes that of Nash equilibrium, which corresponds to the situation where σ^* is the product of N independent distributions, one for each player. In a correlated equilibrium the distributions are not necessarily independent.*

The games we can solve in practice

One reason we are interested in that solution concept is that in 1950, Nash proved in [8] that if a game has N players and each player has a finite number of pure strategies, then this game has at least one Nash equilibrium. As Nash equilibria are also correlated equilibria, this proves the existence of correlated equilibria for the same class of games. Thus if we want to use these results, we need to restrict ourselves to a finite number of pure strategies, i.e., we have to *discretize and bound* the values that each of the variables we defined in Figure [5.1] can take. Even if having done that guarantees the existence of at least one Nash equilibrium, it doesn't tell us how to find it. Actually finding a Nash equilibrium in a normal form game (i.e., a game represented by a matrix in which we store all the payoffs of every player for every pure strategy profile) has been shown to be PPAD-complete [9]. On the other hand there exist some algorithmic approaches that compute correlated equilibria [10] in polynomial time.

The computational complexity of the algorithms associated with the determination of an equilibrium is not the only bottleneck: in practice the memory requirement is also a burden in large normal-form games, the matrix of payoffs growing exponentially with the number of players, etc. Thus in order to overcome this difficulty, we will use some tricks in the algorithmic approach we now describe.

Algorithmic approach

In this section we outline the broad principles of the algorithmic approach we propose for finding an equilibrium in our game. This approach supposes two main things:

- We have discretized and bounded the values that each of the variables can take, i.e., there is only a finite set of possible values for each variable;
- We have access to a solver that finds Nash/correlated equilibria in normal-form games (there are some, e.g. Gambit [11] for the Nash equilibria).

The algorithm we propose has four main steps and is based on [4].

```
Step 1 Compute an initial set of pure strategies $\mathcal{S}_i$ for each player $i$ ;
Step 2 Obtain the normal-form game associated with the enumerated strategies ;
Step 3 Compute an equilibrium of the current normal-form game ;
Step 4 Determine whether there is a player with an incentive to deviate ;
if the deviation incentive is greater than a certain tolerance $\epsilon$ then
    update the normal-form game with new strategies ;
    go back to Step 3
end
else
    return the current equilibrium
end
```

Algorithm 1: Algorithmic approach for finding an equilibrium.

Step 1 There are several possibilities for initializing the set of pure strategies $\mathcal{S}_i$ for player i:

- The current strategies used by Aeroplan and its partners;
- Some strategies that would guarantee a minimum profit for each player;
- Optimal strategies for each player (assuming that this player controls all the variables);
- Equilibrium strategies for some subsets of players.

The algorithm convergence rate depends upon the quality of the initialization: thus finding "good" initial strategies is important. But we have to keep in mind that we do not want to include all the pure strategies (the set X_i) of player i into $\mathcal{S}_i$ from the beginning, since it would be impractical in terms of memory.

Step 2 The normal-form game is simply obtained by computing the utility of each player for any combination of strategies in $\mathcal{S}_1 \times \cdots \times \mathcal{S}_N$ and building the associated multidimensional payoff matrix.

Step 4 After having computed a Nash (resp. correlated) equilibrium x^* (resp. σ^*) in the current "restricted game" on the subset $\mathcal{S}_1 \times \cdots \times \mathcal{S}_N$ of $X_1 \times \cdots \times X_N$, we must verify whether or not it is also an equilibrium in the original game, with all the strategies in X_i allowed for the player i (for each player). If the answer is "yes" x^* (resp. σ^*) is an equilibrium in the "True" game; if not we have to widen the "restricted game" by adding new strategies for the players *(i.e., we have to enlarge the sets $\mathcal{S}_i$).* We will use our verification method to discover *"new interesting strategies"*.

- **Test to verify a Nash equilibrium:** Having obtained x^* and the objective functions f_i for each player i, we compute, for each player i:

  $$\hat{x}_i = \arg\max_{x_i \in X_i} f_i(x_i, x^*_{-i}),$$

  where x^*_{-i} refers to the vector x^* without the ith coordinate. If none of the players gains more than ϵ by choosing $\hat{x}_i$ *(i.e., $\forall i,\ f_i(\hat{x}_i, x^*_{-i}) - f_i(x^*) \leq \epsilon$)*, then we consider that x^* is also an equilibrium for the original game. Otherwise we add the strategy $\hat{x}_i$ to $\mathcal{S}_i$.
- **Test to verify a correlated equilibrium:** Given σ^* *(a probability distribution over $\mathcal{S}_1 \times \cdots \times \mathcal{S}_N$)*, we have to solve the following optimization problem for each player a_i and for each $x_i \in \mathcal{S}_i$.

  $$z_i = \min_{\hat{x}_i \in X_i} \sum_{x_{-i} \in \mathcal{S}_{-i}} \sigma^*(x_i, x_{-i}) f_i(x_i, x_{-i}) - \sum_{x_{-i} \in \mathcal{S}_{-i}} \sigma^*(x_i, x_{-i}) f_i(\hat{x}_i, x_{-i})$$

  If z_i is negative for some value of x_i then σ^* is not a correlated equilibrium for the original game and $\hat{x}_i$ must be added to the set of strategies $\mathcal{S}_i$. On the other hand, if $z_i \geq 0 \quad \forall i\ \forall x_i \in \mathcal{S}_i$ holds, we can return σ^*.

Limitations of the probability-based strategies

The solution concepts we have used in this section were Nash and correlated equilibria. We saw that there always exists at least one of those if we authorize mixed strategies and presented an algorithmic procedure to find them while taking concerns about memory into account. Hence our methodology is expected to return a solution to our problem. This solution might not be a pure strategy profile and indeed there is no guarantee at all that there exists a pure strategy profile. The output of our method could include a recommendation such as *"set $m^{red}_j = 1.6$ for $4/5$ of the time and $m^{red}_j = 15.1$ for $1/5$ of the time"*, which is difficult to implement in practice. Indeed it could be hard to convince players to follow a certain rule based on sampling random numbers. Moreover having a mixed strategy implies that we are trying to maximize an *expected* payoff, but if the game is played a few times only trying to maximize an expectation does not make much sense. Finally we considered a non-cooperative game setting to compute those equilibria *(a compulsory setting for the Nash equilibria)*, but in the real world there may be some sort of cooperation between the players that benefits them.

5.4.2 Notions of cooperative game theory

To remedy the concerns raised above we can place the problem in another setting: a cooperative game. As discussed earlier this setting allows for the search of strategy profiles that maximize the sum of all the agents' payoffs *(the coalition)*. This allows for higher overall payoffs than if we summed the individual payoffs found with the Nash equilibria, where there is competition instead of cooperation. The problem is then to determine how to split this global payoff into shares (one for each player) in a fair and acceptable way.

To solve the problem we can use the notions of *Shapley value* [5] or *core* [6].

- Within the Shapley value framework everybody gets a value "proportional" to the incremental value he (she) brings to the coalition. Moreover the Shapley value always exists and is easy to compute (its downside being that there might be incentives for groups of partners to exit the coalition).
- Within the framework of the core everybody is awarded a payoff guaranteed to be better than what he (she) would have obtained by not being in the partnership. Moreover there is never an incentive for a group of partners to exit. The difficulty with the core is that there is not always a core; in practice, many interesting games have a core.

Computing those values can then give indications to Aeroplan as to what agents have the market power and what the fair rates are for the partners.

5.5 Conclusion and further challenges

During the course of this workshop we focused on modelling correctly the intricate interactions between Aeroplan, its members, and its partners. We enumerated all the actions each of these agents can take that affect the behaviour of others, we formulated a mathematical program (objective functions and constraints) for each agent, and we took into account the specific position of Air Canada in Aeroplan's objective (Air Canada being both the owner and a partner of Aeroplan). Then we discussed how this set of optimization problems can be solved using a game-theoretical approach, highlighting the numerous ways in which "solving" can be interpreted. We explored different solution concepts and outlined an algorithmic approach for computing Nash equilibria, a broadly accepted solution concept. We also gave some insight on what type of solutions each concept would give, their advantages, the impact they would provide, and their limits in a real-world setting.

Nevertheless many research directions require further work. Producing a sound regression function that models well the members' behaviour is still a challenge. Moreover, even if we gave some indications as to how to overcome this challenge, we still need to devise an explicit method to deal with the non-linearity of the regression, which produces nasty feasible sets for our optimization problems. There is still work to be done in order to implement an actual solver for the game at hand.

Bibliography

[1] Ross Anderson, Joey Huchette, Christian Tjandraatmadja, and Juan Pablo Vielma. Strong mixed-integer programming formulations for trained neural networks. In IPCO, 2018.

[2] Huw Dixon. Equilibrium and explanation. In The Foundations of Economic Thought, pages 356–394. Oxford: Blackwells, 1990.

[3] Robert J. Aumann. Subjectivity and correlation in randomized strategies. Journal of Mathematical Economics, 1(1):67–96, 1974.

[4] Margarida Carvalho, Andrea Lodi, and João Pedro Pedroso. Computing Nash equilibria for integer programming games. DS4DM–2018–006, 2018.

[5] Lloyd S. Shapley. A value for n-person games. In Contributions to the Theory of Games (AM-28), Volume II, pages 307–318. Princeton: Princeton University Press, 1953.

[6] Donald B. Gillies. Solutions to general non-zero-sum games. In Contributions to the Theory of Games (AM-40), Volume IV, pages 47–86. Princeton: Princeton University Press, 1959.

[7] Christos H. Papadimitriou and Tim Roughgarden. Computing correlated equilibria in multi-player games. J. ACM, 55(3):14:1–14:29, August 2008.

[8] John F. Nash. Equilibrium points in n-person games. Proceedings of the National Academy of Sciences, 36(1):48–49, 1950.

[9] Constantinos Daskalakis, Paul W. Goldberg, and Christos H. Papadimitriou. The complexity of computing a Nash equilibrium. In Proceedings of the Thirty-eighth Annual ACM Symposium on Theory of Computing, STOC '06, pages 71–78, 2006.

[10] Albert Xin Jiang and Kevin Leyton-Brown. Polynomial-time computation of exact correlated equilibrium in compact games. In Proceedings of the 12th ACM Conference on Electronic Commerce, EC '11, pages 119–126, 2011.

[11] Andrew M. McKelvey, Richard D. McLennan, and Theodore L. Turocy. Gambit: Software tools for game theory, version 16.0.1., 2014.

6 Poisson regression for smooth geographic stratification of risk

Delphine Boursicot[a]

Philippe Gagnon[b]

Rachel Han[c]

Tony Wong[c]

Michael Lindstrom[d]

Nassim Razaaly[e]

Juliana Schulz[a]

Junwei Shen[f]

Maxime Comeau[g]

Bastien Ferland-Raymond[g]

Charles Gauvin[g]

[a] HEC Montréal, Montréal (Québec), Canada

[b] University of Oxford, Oxford, United Kingdom

[c] University of British Columbia, Vancouver (British Columbia), Canada

[d] University of California, Los Angeles, USA

[e] DeFI Team (INRIA SIF, École Polytechnique), France

[f] University of Western Ontario, London (Ontario), Canada

[g] MRCC, Desjardins General Insurance Group, Montréal (Québec), Canada

October 2020
Les Cahiers du GERAD

Abstract: *Segmentation of risk over spatial location is important in the insurance industry, particularly for home insurance, as each region has its own innate level of risk based on features of the location and its surroundings. It is also important that risk segmentation be spatially smoothed for business considerations: that is, models should not predict risk levels that vary rapidly in space, in order to avoid unfair pricing differences for two clients in similar living conditions separated by a short distance. In this report we outline the approaches we took to address this problem. In particular we applied the methods of Geographically Weighted Regression, Poisson Kriging, and Fused Lasso, to insurance claim counts data from Desjardins. The models were applied to aggregated data on a postal code basis in order to predict spatial risk levels.*

6.1 Introduction

The Desjardins *Groupe d'assurances générales* submitted a problem to the 2019 Montréat Industrial Problem Solving Workshop (organized by the CRM and IVADO). The problem was to investigate and develop statistical methodologies capable of characterizing the geospatial elements of the risk of household theft, while ensuring both prediction accuracy and a slow variation of geographical estimators. At the time of the workshop the company was using a four-step process: (i) a piecewise (over each region) generalized linear model (GLM), (ii) supplemented by gradient boosting with Xgboost, (iii) a subsequent smoothing via Markov Random Fields, and finally (iv) a prediction with the resulting smoothed GLM. This methodology was somewhat convoluted, involving many steps, and lacked robustness against small changes in the data. The challenge posed was to identify alternative methods to address the problem specifications.

We investigated three alternatives: Geographically Weighted Regression (GWR) [1], the Kriging Method [3], and Fused Lasso [4], each modified suitably to model Poisson random counts. Geographically Weighted Regression is a technique that provides a local regression for a response variable at every point in space, with nearby observations having a greater influence in the regression: this is similar to a locally weighted scatterplot smoothing (LOWESS) approach. The Kriging method bears similarities to GWR but in making a prediction, empirical spatial autocorrelations in the response variable are taken into account to develop the weightings. With Fused Lasso, a regression is obtained for the data in conjunction with two penalty terms, the first being used for model selection (reducing parameters) and the second for enforcing spatial smoothness of estimators.

Having described the methods, the remainder of this report is organized as follows: in Sections 6.2 and 6.3, we introduce our data and notation conventions; in Section 6.4, we explain our techniques; our results are presented in Section 6.5 and we provide a conclusion and discuss future work in Section 6.6.

6.2 Data

The data provided were individual records of insured individuals and their resulting number of claims of household theft. Several hundred features were included in each record, including personal characteristics such as age and geographic features such as local crime. For the sake of privacy no names were included and all data were provided at the postal code level. Several record features were of a categorical nature, such as "type of roof." Close to 900,000 records were provided for analysis.

Because of the enormous size of the dataset in terms of both records and number of variables, we chose to employ various reductions and simplifications. These included the following.

- *One-hot encoding* [5]: Categorical data were represented in a binary form. This greatly increases the number of variables but allows for categorical data to be treated numerically. Otherwise there is no a natural way to combine numerical values and categorical values.

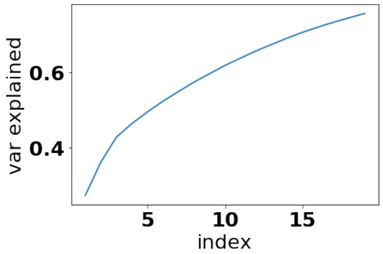


Figure 6.1: Plot of cumulative variance explained versus the number of principal components used after the subsampling of features was performed and one-hot encoding was applied. When PCA was used, we kept the first 10 principal components only.

- *Feature Selection*: For the models we selected 50 features as being the most important and influential and we worked with these 50 features only. This was done with Xgboost [6].
- *Subsampling*: We applied the methods to a dataset of approximately 47,000 records rather than 900,000.
- *PCA*: Principal component analysis [7] allows for high-dimensional data to be represented, at least approximately, in a linear subspace of lower dimension (see Figure 6.1).

In Geographically Weighted Regression, for simplicity we did not convert the latitude and longitude to distances and we treated each degree change in latitude as equal in length to a degree change in longitude. For making actual predictions, proper conversions would be in order.

For the Kriging approach we used the R package geoRglm to carry out the analysis; this did not allow us to incorporate a weight variable for each location, affecting the level and validity of the predictions. This difficulty need to be addressed.

6.3 Notation

We present here the notation used in describing the models. In general we assume a features matrix

$$\Theta \in \mathbb{R}^{N\times(1+d_c+d_s)},$$

where there are N records of clients, with d_c client-specific features (age, etc.) and d_s geography-specific features (local crime rate, etc.). The extra 1 is for the intercept. Thus the first column of Θ is an all ones vector, columns 2 through $1+d_c$ describe the client data, and columns $2+d_c$ through $1+d_c+d_s$ describe the spatial data. We shall adopt the notation that a subscript of *i:* denotes the ith row of a given matrix. We also use

$$X \in \mathbb{R}^{N\times 2}$$

to denote a matrix of N corresponding locations (centroids of postal codes in our data). We also use a *column* vector

$$y \in \mathbb{R}^{N}$$

to denote the realizations (claim counts).

Some models encoded spatial location as one of the d_s spatial variables. Sometimes in this report we will use

$$T \subset \{1, 2, ...N\}$$

to denote the set of records used for training and

$$V \subset \{1, 2, .., N\}$$

a set of records used for validation. We often required that the centroids contained in the two sets be disjoint, i.e.,

$$(\cup_{i\in T} X_{i:}) \cap (\cup_{i\in V} X_{i:}) = \emptyset. \tag{6.1}$$

This is relevant because proper validation and parameter selection require a fair test of the models' predictive powers: i.e., there should be no biasing of the models with samples of regions it needs to predict. For the Fused Lasso method, however, locations are categorical data and the disjointedness condition in (6.1) is removed.

As our models are predictive ones, a new client (not part of the known records) could be assigned a record index $i \notin T \cup V$ and our models could make predictions as to their claim rate using additional data $\Theta_{i:}$ and a position $X_{i:}$.

We interpret Y_i, the number of insurance claims made by client i, as a Poisson random variable where $\Pr(Y_i = y_i)$ can be explicitly computed from a known Poisson rate λ_i.

$$\Pr(Y_i = y_i|\lambda_i) = \exp(-\lambda_i)\frac{\lambda_i^{y_i}}{y_i!} \tag{6.2}$$

We apply a GLM to describe the Poisson rate so that client i is modelled as having $Y_i \sim \mathrm{Pois}(\lambda_i)$, where

$$\log \lambda_i = \Theta_{i:}\hat{\beta}_i + \varepsilon_i. \tag{6.3}$$

In the GWR and Lasso methods we have $\varepsilon_i = 0$, but $\varepsilon_i \neq 0$ holds for the Poisson Kriging method. The vector $\hat{\beta}_i$ is a column vector of estimators to be determined for location $X_{i:}$: the way to determine $\hat{\beta}_i$ depends upon the method chosen.

6.4 Techniques

6.4.1 Geographically Weighted Poisson Regression

Method

To make a prediction for a client index i (possibly not in T), with a known feature row-vector $\Theta_{i:} \in \mathbb{R}^{1+d_c+d_s}$ and location $X_{i:}$, we compute

$$\hat{\beta}_i = \mathrm{argmin}_\beta \left(-\sum_{j\in T} w(X_{i:}, X_{j:}) \log \Pr(Y_j = y_j|\lambda_j = \exp(\Theta_{j:}\beta)) \right), \tag{6.4}$$

which is a weighted log-likelihood of observing all of the data. In general a w function is chosen to be a decreasing function of the distance between its arguments. For our work we chose the Gaussian

$$w(x, y) = \exp\left(-\frac{||x-y||^2}{2\alpha^2} \right), \tag{6.5}$$

where x and y are two positions. Other choices can be made. The value of α in (6.5) is a hyperparameter, which we chose by cross-validation. For GWR, we used the top 10 principal components in developing the regressions.

Evaluation

To perform cross-validation and assess the model fits, we considered two objective functions $J(\alpha)$. If V denotes the set of points in the validation set, i.e., all clients at a point with a Poisson rate to be estimated, we used

$$J(\alpha) = L(\alpha) = -\frac{1}{|V|}\sum_{i\in V} \log \Pr(Y_i = y_i|\lambda_i), \tag{6.6}$$

the average negative log likelihood of observing the validation set (see (6.2)), and

$$J(\alpha) = D(\alpha) = \frac{1}{|V|}\sum_{i\in V}\left(y_i \log(y_i/\lambda_i) - (y_i - \lambda_i)\right), \tag{6.7}$$

the average deviance. It is understood that if $y_i = 0$, the logarithm term is not included as part of the summand in (6.7). The optimal α, denoted by α^*, is given by

$$\alpha^* = \text{argmin}_\alpha J(\alpha). \tag{6.8}$$

In Figure 6.2, the results of the objective functions for different values of α are plotted.

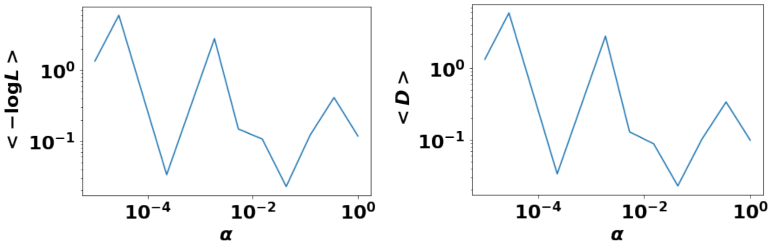


Figure 6.2: A parameter sweep over α for the objective functions (6.6) and (6.7) to estimate the best hyper-parameter α. The values plotted are the mean values over the validation set. The optimal value is $\alpha \approx 4.3 \times 10^{-2\circ}$.

6.4.2 Poisson Kriging

To implement Poisson Kriging we made the assumption that the log of the Poisson rate could be expressed as the sum of an average rate that can be predicted from the features and a Gaussian random field. This resulted in

$$\log \lambda_i = f(X_{i:})\hat{\beta} + \varepsilon(X_{i:}) \equiv s_i, \tag{6.9}$$

where $f(X_{i:})$ is a row vector of geovariables for the i^{th} location plus the intercept term of 1. In the above equation $\hat{\beta}$ is an estimator to be found by maximizing the marginal likelihood and ε is a random field with mean $\mu = -c(0)/2$ and covariance function

$$C(x, y) = c(||x - y||) = \sigma^2 \exp(-||x - y||^2/\phi),$$

with σ and ϕ as parameters. So when $|c(0)|$ is small a Taylor expansion yields the following.

$$\begin{aligned}
E(Y_i) &= E[E(Y_i|\varepsilon(X_{i:}))] \\
&= e^{\Theta_{i:}\hat{\beta}} E\left(e^{\varepsilon(X_{i:})}\right) \\
&\approx e^{\Theta_{i:}\hat{\beta}} \left(e^{\mu} + \frac{1}{2}e^{\mu}\text{Var}(\varepsilon_i)\right) \\
&= e^{\Theta_{i:}\hat{\beta}} e^{-c(0)/2}\left[1 + \frac{1}{2}c(0)\right] \\
&\approx e^{\Theta_{i:}\hat{\beta}} e^{-c(0)/2} e^{c(0)/2} = e^{\Theta_{i:}\hat{\beta}}
\end{aligned}$$

To estimate σ^2 and ϕ, we estimated the spatial covariance of the response variables Y_i by empirically estimating the semivariogram function (the relationship between the semivariance of a response variable and the distance between points).

$$\gamma(r) = \frac{1}{2}\text{Var}(\{Y(u) - Y(v) \text{ s.t. } |u - v| = r\}) \tag{6.10}$$

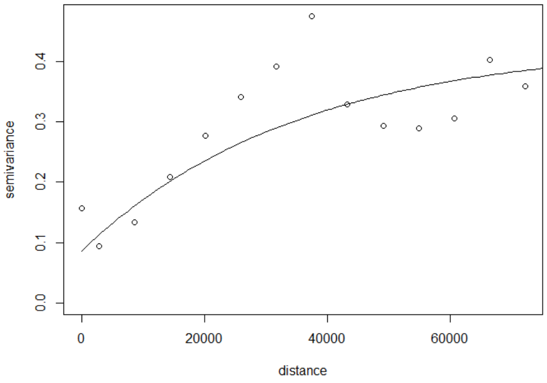


Figure 6.3: Empirical semivariogram plot based on all the available data. The sill (σ^2), range (ϕ), and nugget (τ^2) parameters can be estimated approximately from this plot: they are respectively around 0.37, 38,000, and 0.08.

Equation (6.10) describes the variance in the difference between the response variable values at any pair of points a distance r apart. In theory it should vanish at $r = 0$ and increase as r increases, possibly saturating at some point. In practice the continuous values of r are replaced by buckets of finite width. This curve is related to the covariance function through the relation $c(r) + \gamma(r) = \gamma(\infty)$, so that knowing one of the two functions allows the other to be found as well. The fit we obtained for the semivariogram is illustrated in Figure 6.3. In practice a nugget is present (nonzero y-intercept) and the semivariance is nonzero at zero distance. This is due to possible measurement errors during the collection of data.

After computing the semivariogram we were able to estimate the parameters σ^2 and ϕ in the covariance function. Suppose $Y = (y_1, y_2, \ldots, y_n)^T$ is the response for the training data and $Y^* = (y_1^*, y_2^*, \ldots, y_t^*)^T$ is the response for the test data. We also defined S, S^*, ε, and ε^* as the respective vectors $(s_1, ..., s_n)^T$, $(s_1^*, ..., s_t^*)^T$, $(\varepsilon_1, ..., \varepsilon_n)^T$, and $(\varepsilon_1^*, ..., \varepsilon_t^*)^T$. Here we make the prediction of y^* through the prediction of S^* and ε^*. We can predict ε^*, the $\varepsilon_i^{*\prime}s$ at positions $X_{1:}^*, \ldots, X_{t:}^*$, thanks to information from ε. Then predictions can be made via MCMC simulation as described in the following algorithm.

Step 1: Simulate ε from the distribution of ε conditioned on y:

$$P(\varepsilon|Y) \propto P(Y|\varepsilon)P(\varepsilon)$$

where

$$P(Y|\varepsilon) = \prod_{k=1}^{n} \frac{\exp(s_k y_k)}{y_k!} e^{-\exp(s_k)},$$

$$s_k = \hat{\beta} f(X_{k:}) + \varepsilon(X_{k:})$$

and ε comes from the Gaussian random field with parameters $\hat{\sigma}^2 = \sigma^2$ and $\hat{\phi} = \phi$ estimated from the semivariogram.

Step 2: Simulate ε^* from the distribution of ε^* conditioned on ε (since they come from the same Gaussian random field, the conditional distribution is also Gaussian with associated parameters $\hat{\sigma}^2$ and $\hat{\phi}$).

Step 3: $s_k^* = f(X_{k:}^*)\hat{\beta} + \varepsilon^*\,(X_{k:}^*)$ and $y_k^* = e^{s_k^*}$.

Step 4: Repeat steps 1–3 m times, where m is the number of simulations. Then the final predicted response for the t locations of interest is the average value of the m predictions.

6.4.3 Fused Poisson Lasso

In the Fused Poisson Lasso regression, the spatial location (i.e., postal code) is represented by a categorical variable with d_ℓ values and encoded by a set d_ℓ binary covariates. Denote by $\mathcal{L}$ the set of column indices containing these covariates. We seek a single vector of estimators $\hat{\beta}$ that is the solution of the following minimization problem.

$$\begin{aligned} \hat{\beta} &= \mathrm{argmin}_\beta \left\{ -\frac{1}{|T|} \sum_{i \in T} \log \Pr(Y_i = y_i | \lambda_i = \exp(\Theta_i \beta)) \right\} \\ \text{subject to} \quad & \sum_j |\beta_j| \leq t_1 \text{ and } \sum_{j \in \mathcal{L}} |\beta_j - \beta_{j^*}| \leq t_2 \end{aligned} \tag{6.11}$$

In the above equation $t_1, t_2 > 0$ denote hyperparameters that require tuning and j^* is the index of the "nearest" (in a sense defined below) neighbor of the j^{th} location. Note that the penalization above represents a special case of a broader penalization framework, where for instance the differences between all coefficients of the neighbours of the j^{th} location would be explicitly penalized.

The ℓ^1-constraint with t_1 as the right-hand side is a model selection mechanism, chosen so as to make the regression coefficients in $\hat{\beta}$ sparse. The ℓ^1-constraint with t_2 as the right-hand side was chosen so as to make the (categorical) coefficients describing similar locations identical (if it makes sense in the context of the optimisation problem): that is, if a client moves from region 1 to region 2, where both regions have similar geographic features (median income, local crime, etc.), and his (her) successive houses are in similar conditions, there should not be a significant change in his (her) risk factor.

In practice, given a location indexed by j, we defined j^* to be the index of the "nearest" location in terms of distance with respect to the features. Because of the time limitation we used Euclidean distance in the feature space.

Unlike GWR, this method explicitly assumes that regardless of spatial location (for all values of the categorical variable *location*), the other covariates all have the same effect upon the response variable. Regardless of whether this is true or not, the assumption corresponds to the problem specifications (where client-specific factors cannot have an influence over space).

6.5 Results

Here we present the results of the different techniques.

6.5.1 Geographically Weighted Poisson Regression results

In Figure 6.4, with the optimal α, we plot the prediction of the Poisson rate λ at a subset of locations. One can observe a degree of smoothness in the risk.

6.5.2 Poisson Kriging results

Using Poisson Kriging as described, we can generate a prediction for the frequency of claims being made over space as seen in Figure 6.5.

6.5.3 Fused Poisson Lasso results

In Figure 6.6 we display the spatial distribution of the risk obtained through the Fused Lasso method with various values for t_1 and t_2.

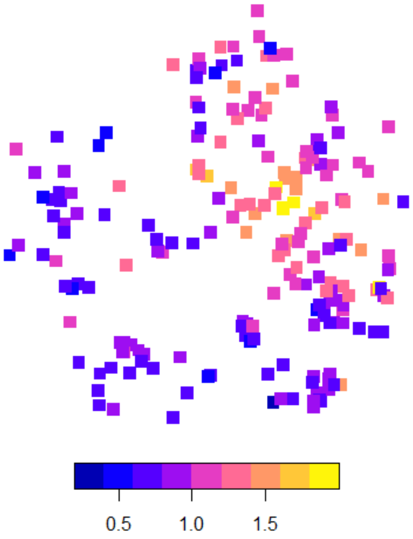


Figure 6.4: Over a small validation set the average predicted λ for GWR is 0.00644 per year and the plotted values are relative to this average value.

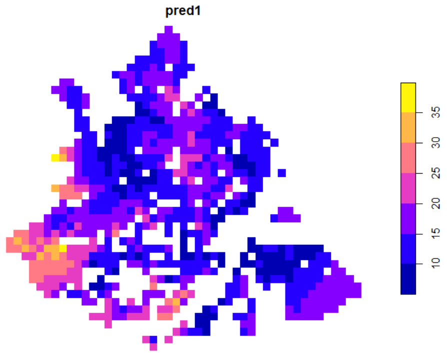


Figure 6.5: Plot of risk over space using Poisson Kriging. The response variable is smooth. It is important to note a limitation in the R package geoRglm**, which is not able to manage weights in observations; actual frequencies are therefore not accurate.**

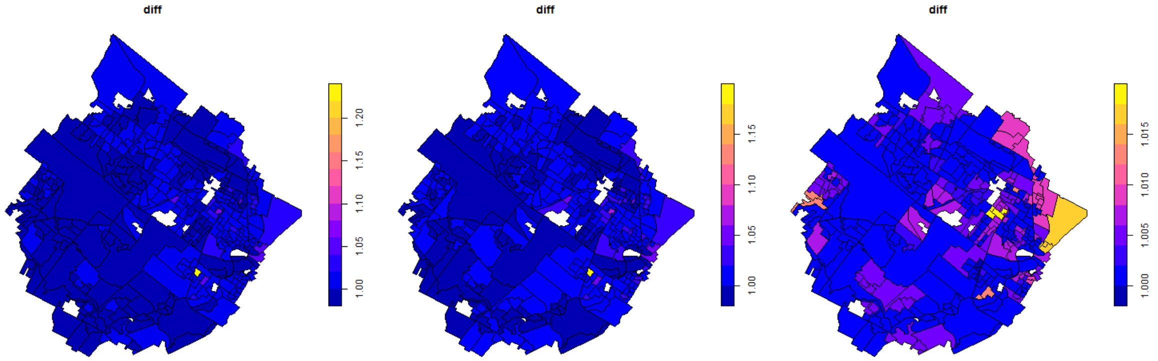


Figure 6.6: Plot of risk over space using Fused Lasso. The different plots show the results for different choices of t_1 and t_2 (the values increase as we move to the right).

6.6 Conclusions and future work

Through the workshop we built upon and used three different techniques for the prediction of risk over space. All techniques were able to provide predictions of risk over space, and each method has its own pros and cons. We identify them below.

For Geographically Weighted Regression

- A single model serves to describe the entire dataset and make predictions;
- The resulting spatial distribution of predicted risks appears to be smooth;
- Predictions can be made at locations where there were no previous data;
- The algorithm is flexible in that different forms of weighting functions could be used, but this also adds a level of arbitrariness to the fitting;
- The algorithm is time-consuming. It took more than two hours to run a sweep of 15 α-values in cross-validation using only 1% of the unique postal codes in the smaller dataset of 47,000 records.

For Poisson Kriging

- The basic model is very simple and easy to interpret;
- Smoothness emerges from using the model;
- Unfortunately the model is difficult to apply for Poisson and other distributions (such as Gamma and Tweedie, which are often used in insurance to model individual claim amounts and total losses, respectively);
- The algorithm is slow and consumes a lot of memory: running it took more than two hours and 25G of RAM for an input of 5000 records.

For Fused Lasso

- The method is "global," fitting everything all at once, and the results are intuitive;
- It meets the problem specifications;
- It cannot directly make predictions at locations where no client data are available since that would introduce a new categorical value. For a new location, however, heuristics based on finding the most similar location from known observed points (in a manner akin to the use of nearby locations by GWR) could be used;
- It can be very memory-intensive: to make a prediction one requires at least as many dummy variables as there are unique client locations. We can, however, *fuse* areas together (the j-th location with its nearest neighbour, for instance), and consider the result as a new area with its own nearest neighbour prior to running the optimization, until the number of covariates is reasonable.

In future work it would be worthwhile to consider the following:

- Modify the objective function in the GWR to perform model selection (by adding an ℓ^1-penalty to the minimization problem (6.4));
- Improve the management of observation weights in Poisson Kriging and increase the amount of data that can be analyzed;
- Use the Mahalanobis distance [2] to select nearest neighbours for Fused Lasso.

Author contributions

All authors contributed to the scientific investigations, wrote code, and interpreted and discussed results. The report was written by Michael Lindstrom with technical assistance from Philippe Gagnon and Juliana Schulz. All authors read and reviewed the final manuscript.

Bibliography

[1] A Stewart Fotheringham, Chris Brunsdon, and Martin Charlton. Geographically weighted regression: the analysis of spatially varying relationships. John Wiley & Sons, 2003.

[2] Prasanta Chandra Mahalanobis. On the generalized distance in statistics. National Institute of Science of India, 1936.

[3] Mohammad Ali, Pierre Goovaerts, Nushrat Nazia, M Zahirul Haq, Mohammad Yunus, and Michael Emch. Application of Poisson kriging to the mapping of cholera and dysentery incidence in an endemic area of Bangladesh. International journal of health geographics, 5(1):45, 2006.

[4] Robert Tibshirani, Michael Saunders, Saharon Rosset, Ji Zhu, and Keith Knight. Sparsity and smoothness via the Fused Lasso. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 67(1):91–108, 2005.

[5] Joseph E Beck and Beverly Park Woolf. High-level student modeling with machine learning. In International Conference on Intelligent Tutoring Systems, pages 584–593. Springer, 2000.

[6] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pages 785–794. ACM, 2016.

[7] Hervé Abdi and Lynne J Williams. Principal component analysis. Wiley interdisciplinary reviews: Computational statistics, 2(4):433–459, 2010.

[8] George Kour and Raid Saabne. Real-time segmentation of on-line handwritten arabic script. In Frontiers in Handwriting Recognition (ICFHR), 2014 14th International Conference on, pages 417–422. IEEE, 2014.

[9] George Kour and Raid Saabne. Fast classification of handwritten on-line arabic characters. In Soft Computing and Pattern Recognition (SoCPaR), 2014 6th International Conference of, pages 312–318. IEEE, 2014.

[10] Guy Hadash, Einat Kermany, Boaz Carmeli, Ofer Lavi, George Kour, and Alon Jacovi. Estimate and replace: A novel approach to integrating deep neural networks with existing applications. arXiv preprint arXiv:1804.09028, 2018.

7 Prioritizing abnormal situations automatically in the context of fraudulent claims

Anas Abdallah[a]
Helen Samara Dos Santos[b]
Caio De Naday Hornhardt[b]
Francis Duval[c]
Anthony Forgetta[d]
Matthew Griffith[e]
Manuel Morales[f]
Lindon Roberts[g]
Fabian Ying[g]
Bowei Zhang[h]
Steven Côté[i]
Marc-André Desrosiers[i]
David Mazurkiewicz[i]

[a] McMaster University, Hamilton (Ontario), Canada

[b] Memorial University, St. John's (Newfoundland), Canada

[c] Université du Québec à Montréal, Montréal (Québec), Canada

[d] Concordia University, Montréal (Québec), Canada

[e] University of Bath, Bath, United Kingdom

[f] Université de Montréal, Montréal (Québec), Canada

[g] University of Oxford, Oxford, United Kingdom

[h] University of Western Ontario, London (Ontario), Canada

[i] The Co-operators, Canada

October 2020
Les Cahiers du GERAD

Abstract: *When processing an indemnization claim in the insurance industry, it is critical to determine quickly whether the claim is potentially fraudulent or not. This enables one to choose the actions needed in the investigation and to slow down the payment process in order to allow the investigation to take place. Since human judgments often display a lack of consistency, one would like to design automated prediction processes that will identify a preliminary group of situations involving fraud. The common experience of the insurance industry suggests that it is crucial to take into account networks of relationships among individuals, including their diverse roles, addresses, vehicles, etc., especially when fraudulent activities take place within these networks. Generally speaking, the investigative teams in the insurance industry cannot process all the leads they receive: there is an urgent need for more "intelligence" in the automated models selecting the leads submitted to investigators. Given (a) the relations known to the insurance company, (b) the informations on the client, the insurance policy, and the claim details available at the time the claim is filed, (c) the annotations made by the investigating team before the filing of the claim (there may be up to 1000 annotations over a period of six years), the problem consists of designing a model that predicts whether a claim is fraudulent or not. Once the model has been proposed, its quality must be evaluated. In order to do so the company representatives have provided the team with a set of new claims and examined the claims prioritized by the model.*

7.1 Data

To achieve our goals, we have at our disposal a claim dataset containing information about both fraudulent and non-fraudulent claims, which includes structured fields about the household, their policies, features of the risk, and features of the claim. It also includes elements commonly examined by investigators, such as the time since the policy took effect and the time of day of the claim. In addition there is network information, the purpose of which is to make the links between the different entities (claim, claimant, policy, driver, vehicle, household, etc.) within a claim file. In the claim dataset there are three types of claims annotations (reflecting the risk of fraud) made by human experts. Claims that have been successfully demonstrated fraudulent are labelled "golden standard," those that have been referred for suspicion of fraud but for which the fraud has not been proven are labelled "silver standard," and claims that are referred for opinion are labelled "bronze standard." Out of about 50,000 claims, around 30, 60, and 60 are annotated golden, silver, and bronze (respectively). The data focuses on areas outside the greater Toronto area in Ontario and on insurance of physical persons.

7.2 Network representation for fraud identification

The aim is to identify fraudulent car insurance claims using network analysis. The main idea is to identify groups of claims that are connected to one another (e.g., by sharing an address), as these may be suspicious. To do this, we create networks of claims. We connect claims by an edge if they share a common property (e.g., the same address or the use of the same repair shop). We examine whether networks created in this way can help identify fraudulent claims.

In particular, we examine two networks:

1. *The Personal Network*: We connect claims if they share any of the following properties:
   (a) Any person on the claim (driver, policy holder, claimant, main contact person, payee): two claims are connected if any person appears in both claims (e.g. driver on claim 1, payee on claim 2)
   (b) Address
   (c) Policy
   (d) Household

(e) Note: we believe including other connections, such as shared phone numbers or email addresses, would also provide useful network information. This data is not currently available, however.

2. *The Vendor Network*: We connect claims if they are associated with a common third party:
   (a) Vendor (e.g., repair shop, clinic)
   (b) Insurance advisor

The personal network The goal of the personal network is to identify groups of people who in all likelihood know one another (e.g. have a shared address or appear on a policy) and have all appeared on claims. Our hypothesis for the personal network is that claims that are linked (through the personal network connections) to many other claims are more likely to be suspicious. There are several ways of measuring the strength of connections of a vertex within a graph:

1. Vertex degree: that is, how many other claims is a particular claim directly linked with?
2. Size of connected component: how many claims is a particular claim linked with in total, including via indirect connections (e.g. Claim 1 is linked with Claim 2, which is then linked with Claim 3)?

The vendor network The goal of the vendor network is to identify vendors with a higher rate of participation in fraudulent claims. We cannot use the vendor network within the personal network, because there are many vendors who are legitimate, but have many associated claims (e.g. a large repair shop in a major city). Therefore the fact that claims are strongly linked through the vendor network is not necessarily a risk factor. We consider instead two ways of identifying fraud through the vendor network (there may be other ways):

- Claims that are closely connected (through vendors) to known fraudulent claims are more likely to be fraudulent. This is a “supervised” approach, as it requires a list of known fraudulent/suspicious claims.
- Claims that are closely connected (through vendors) to potentially “suspicious” claims, as measured by some approach not using labelled frauds, such as their score from the personal network.

In both instances, we expect that some/all claims (vertices in the graph) will have an associated “risk score” (e.g. 0 if there is no fraud, 1 if there is a fraud, or a personal network “suspicion” measure). With this information, vertices within the vendor network may receive a score through several measures, such as:

1. Distance to fraud: a measure of how far away (via vendor connections) is a particular claim from a known fraudulent claim. In this case it may make sense to treat claims with multiple vendors in common as “closer” than claims with only one vendor in common.
2. Average “risk score” of nearby claims (including all claims that are at most X links away from a particular claim, where X is small, e.g. $X = 1$ or $X = 2$).

Use of network information Ultimately this methodology could lead to the following workflow.

1. Maintain a network of claims (as described above), which is regularly updated as claims are added or details changed.
2. When a new claim arises, include it into the relevant network and compute relevant scores (e.g. measures of “connectedness” within the personal network).
3. Use the scores to determine whether or not to escalate. There are several ways in which this could be done: if the score is high compared with those of other claims (e.g. in the top 5%), compute a combined “risk score” from several measures or even incorporate any score(s) as a feature in a supervised learning/regression model.

There may be other simple rules that are relevant for making an escalation decision, such as value of claim (or total value of linked claims) or location (e.g. rural areas have a higher risk of animal collisions).

Example of a personal network To demonstrate the benefits of the approach just described, we used four years of auto claims data to build a personal network. Based on the available data, we connected claims based on shared policy, named person (claimant, policy holder, etc.), address, or "household ID." This gave us a network with approximately 32 000 claims, of which around 100 were flagged as potentially fraudulent.

In Figure 7.1, we show a subgraph of the full personal network, with claims flagged as fraudulent shown in red. As expected, we see many connected components (corresponding to claim subsets unrelated to one another): most of these components consist of very few claims. There are, however, a small number of components with several claims linked to one another.

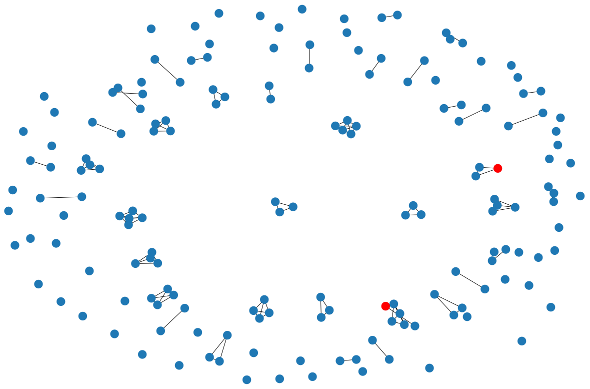


Figure 7.1: Subset of a personal network graph. Red nodes are those that have been flagged as fraudulent.

Within this network we consider a subset of claims that are connected to one another, in order to determine if this provides a useful indicator of fraud risk. Note that we are not so concerned about identifying claims that are likely to be fraudulent: we wish rather to identify claims that *have features making them worthy of escalation to SIU/human intervention.*

In Figure 7.2 we show the largest connected component of the personal network graph. This component consists of 15 claims, linked for a variety of reasons (shared policy, shared address, etc.). Upon detailed inspection by SIU experts from The Co-operators, they concluded that this collection of claims has low fraud risk, and they would not recommend further investigations. In addition some of these edges are likely spurious artefacts of the data extraction process. They note, however, that this pattern of interrelated claims is of interest to them and having automated escalation of these would be useful in practice.

We identified other large connected components, which were also inspected by SIU. In all cases either these components included known fraudulent claims (or claims under investigation), or they included features making the claims worthy of escalation/human consideration. As a result we conclude that network analysis is a promising avenue of investigation for The Co-operators for identifying potentially fraudulent claims.

We note that there exist commercial tools that can perform similar network analysis of claims, aggregating data across multiple insurers. The benefits of The Co-operators having an in-house system based on the approaches we have considered would include: no restrictions on data access (as all claims belong to one company); customization of scoring/alerts; potential to incorporate multiple insurance categories (e.g. home and auto) and multiple regions (as opposed to the current tools, which are restricted to a single province).

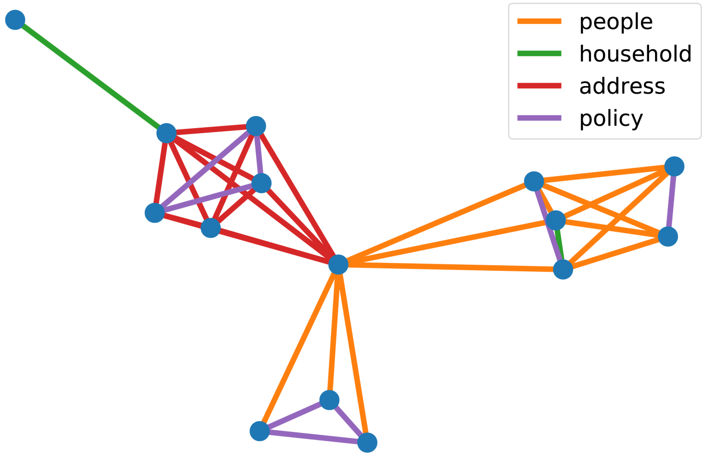


Figure 7.2: Largest connected component of a personal network graph. The edges are coloured by the type of connection they represent (e.g. claims on the same policy, or with a person in common).

7.3 Outliers detection using an autoencoder

The aim of this section is to identify fraudulent claims using an autoencoder. An autoencoder is a type of neural network that performs unsupervised learning. The main idea is to use autoencoders to detect outliers, which will alert the advisor to novel patterns in the data indicating potential frauds.

In particular an autoencoder will learn to identify non-fraudulent claims. It carries out this task by creating a stochastic representation of the original input, with the goal of minimizing a certain distance between the input and its corresponding reconstruction. The Kullback-Leibler divergence is used to measure this distance. The autoencoder consists of two parts: an encoder function $e = f(x)$ and a decoder function $d = g(e)$. To prevent the autoencoder from setting $g(f(x)) = x$ for every x, autoencoders are restricted in ways that allow them to learn only approximate representations of the training data. If a fraudulent claim is input to the autoencoder, its representation (output) will differ significantly from the input. This is to be expected since the autoencoder is trained exclusively on non-fraudulent claims: hence it will only be able to represent successfully claims that have non-fraudulent characteristics. In addition, since the autoencoder only learns to represent non-fraudulent claims, it allows us to overcome a challenge that is often encountered in anomaly detection: imbalanced datasets.

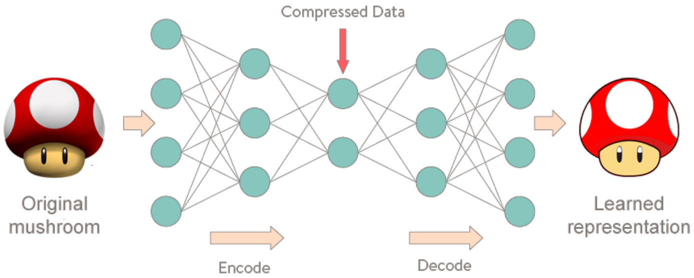


Figure 7.3: Training of an autoencoder.

Specifically less than 1% of our dataset consists of fraudulent claims. The following hyperparameters were manually tuned to optimize the performance of the autoencoder: the number of layers, the number of neurones per layer, the loss function, the optimizer, the learning rate, the dropout rate, the L1 and L2 losses, the batch size, and the number of epochs. Also batch normalization was needed to eliminate the vanishing gradients problem. To measure performance we use recall, precision, and F-measure.

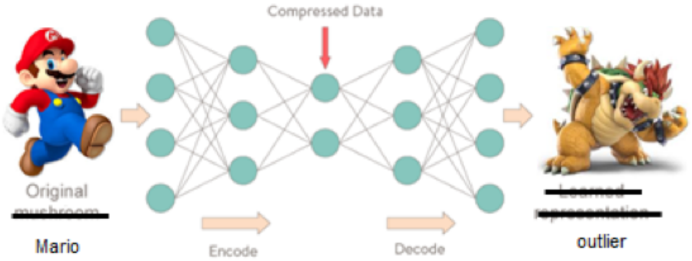


Figure 7.4: How an autoencoder works once it has been trained.

Recall is the proportion of fraudulent claims correctly identified by the autoencoder, whereas precision measures the proportion of fraudulent claims that were correctly predicted. In particular,

$$\text{Recall} = \frac{\text{Number of Fraudulent Claims Identified as Fraudulent}}{\text{Number of Fraudulent Claims}},$$

$$\text{Precision} = \frac{\text{Number of Fraudulent Claims Identified as Fraudulent}}{\text{Number of Claims Identified as Fraudulent}},$$

$$\text{F-measure} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}.$$

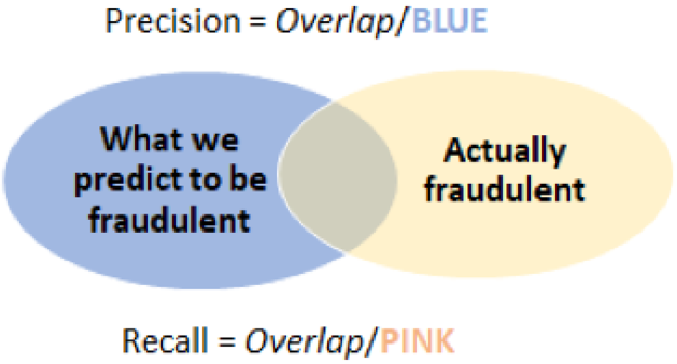


Figure 7.5: Illustration of precision and recall.

Classification algorithms must always make a compromise between precision and recall. Indeed the improvement in precision is often carried out at the expense of recall (which decreases as precision improves) and vice versa. In this particular fraud detection problem we favour a higher recall at the expense of reduced precision. We do this because we do not want to miss any fraudulent claim. Hence we incur the cost of classifying more non-fraudulent claims as fraudulent. We do not consider this to be a problem: false positives is a price that we are willing to pay to avoid false negatives.

Our results demonstrate that the autoencoder can achieve a recall of 100% and a precision of 96% on a test set.

7.4 Supervised learning for fraud detection

In this approach, we try to detect fraudulent claims using machine learning, more specifically supervised learning. To achieve this we have at our disposal a database containing about 100 features of many claims as well as a label for each of them. This labelling has been carried out manually by an expert and reflects qualitatively the likelihood of a claim being fraudulent. There are four kinds of labels: the rust standard, the bronze standard, the silver standard, and the gold standard. Claims labelled with gold are almost surely frauds, while claims labelled with silver and bronze are suspected of being fraudulent. A rust label means there is a small chance of the claim being fraudulent. In order to reach our ends we decided to perform two-class classification on the claim dataset: for this purpose we created an outcome variable taking the value 1 if the label is either gold, silver, or bronze and 0 otherwise. We excluded the rust standard because it is very uncertain whether the claim is fraudulent or not.

The algorithm we chose to perform classification is a forest of classification trees, known as *random forest*. This specific model was retained because it usually yields good prediction results on structured data. This kind of algorithm is often referred to as a black box, which means it is not very good for inference. Hence we focus here on having good predictions and we do not try to understand these predictions.

Classification trees are valued for their simplicity and ease of interpretation but they are not very accurate, mainly because of their lack of robustness. Indeed decision trees suffer from a high variability, meaning that a small change in the data can change the resulting model dramatically. A random forest algorithm consists of building many trees on modified versions of the data. Each tree has a large depth in order to have low bias but high variability. We then get rid of variability by averaging the predictions of all trees.

The database contains about 45 000 claims of which about 200 have an outcome of 1. Because the data is highly unbalanced (i.e., the proportion of outcomes equal to 1 is very small), we perform undersampling, meaning that we get rid of many observations with an outcome of 0 in order to rebalance the classes. After undersampling, we are left with a "outcome of 1" proportion of about 10%. We split the database into a training set and a validation set. The idea is to build classification trees until the out-of-bag error stops decreasing, as illustrated in Figure 7.6. Each tree is fitted on a modified version of the data, that is to say on a bootstrap sample. Also $p = 30$ features are sampled at each iteration. This feature distinguishes the random forest algorithm from the bagging algorithm, where all features are kept. This tweaking allowed us to decorrelate the trees and hence to achieve a better performance. We evaluated the trained model on the validation dataset and we obtained an AUC of 0.91. This looks pretty good, but it is an AUC that is usual for the domain of fraud detection. Using feature engineering would surely have improved the model. The ROC curve obtained is displayed in Figure 7.7.

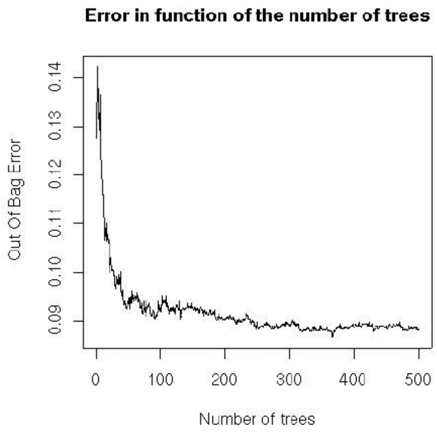


Figure 7.6: When training a random forest algorithm, we stop fitting decision trees when the out-of-bag error stops decreasing. For the random forest algorithm illustrated here, we would choose to keep about 300 trees in the model. Note that the number of trees chosen is subjective.

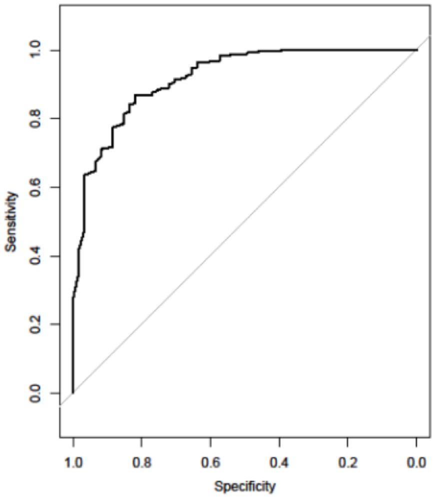


Figure 7.7: ROC curve obtained on the validation dataset.

7.5 Acknowledgements

Lindon Roberts and Fabian Ying acknowledge the support provided by the EPSRC Centre for Doctoral Training in Industrially Focused Mathematical Modelling (EP/L015803/1).

8 Development of a mathematical framework to represent a 2D/3D smooth geometric structure

David Bernier[a]
Dominic Desjardins Côté[a]
Reza Ghamarshoushtari[a]
Scott Gigante[b]
Yuri Grinberg[c]
Mohsen Kamandar Dezfouli[c]
Matthew Hirn[d]
Liangchen Liu[e]
Michael Perlmutter[d]
Paul Sinz[d]
Guy Wolf[f]

[a] Université de Sherbrooke, Sherbrooke (Québec), Canada

[b] Yale University, New Heaven, Connecticut, USA

[c] National Research Council of Canada

[d] Michigan State University, East Lansing, Michigan, USA

[e] University of British Columbia, Vancouver (British Columbia), Canada

[f] Université de Montréal, Montréal (Québec), Canada

October 2020
Les Cahiers du GERAD

8.1 Introduction

The goal of this project from a data processing and machine learning perspective is to generate shapes with “good performance,” where good performance is determined by the underlying physical problem, such as photonic component efficiency.

The photonic component design has been largely driven by manual structure design with a handful of degrees of freedom (parameters). Some method of parameter tweaking or sweeping in conjunction with the physical simulation is subsequently used to identify the right combination of parameters (i.e., one that maximizes the performance). Over the last decade there has been a trend to relax the structural constraints of the design space and let the machine search for optimized designs, mainly using various global search techniques or gradient-based topological optimization methods. The goal is two-fold: existing photonic components can be replaced by components with much smaller footprint without compromising on the performance; components with complex functionalities can be designed without the need to rely on a strong physical intuition (which is not always available). Those approaches, however, still fall short, since typical optimization methods produce only a handful of optimized designs, limiting the understanding of the general high-dimensional parameter space and its capabilities. In particular, since the design space is under-constrained (e.g., many geometries will satisfy the objective), we expect that there exist large pockets of good designs that standard optimization approaches cannot map. Characterizing the subspace of good designs gives a more complete picture to the designer and can help develop physical insights into the geometry. Also such a characterization allows the flexibility to choose the appropriate design, that is, one satisfying other requirements that change from application to application (i.e., different photonic circuits used for different purposes might have their own constraints in addition to the main Figure of Merit) or requirements dictated by different foundries.

As an example we consider the parameterized power splitter design problem https://apps.lumerical.com/inverse-design-y-branch.html. The objective is to design a splitter that splits the power perfectly (50%-50%) between two channels. The shapes are determined by a 10-dimensional vector of parameters, which are optimized relative to the performance objective function: this function is called the Figure of Merit (FoM). The shape is generated from this 10-dimensional vector. A primary aim is to be able to generate shapes in a parameter-free fashion, which will enable more complex and un-intuitive (at least by human standards) designs. In this document we describe a possible machine learning path for this problem, identifying the overall approach, key sub-problems, as well as potential pitfalls and solutions.

The report is organized as follows: the second section describes a design problem faced by NRC researchers; the third section outlines a machine learning approach for this problem (and possibly other design problems); the fourth section examines tools used in the “discriminator” part of the machine learning approach; the fifth section contains a discussion of the “generator” part of this approach; and the remaining sections present ideas based on topological data analysis that could be used in the “generator” and “discriminator” parts of the machine learning approach.

8.2 Two-channel splitter dataset

The dataset provided for the proof-of-concept within this workshop is a set of results from iterative optimizations run from a total of 494 random 10-parameter initializations of the power splitter curve. For each vector of parameters, we observe the gradients from Maxwell’s equations as well as the Figure of Merit denoting the efficiency of the power splitter derived from these equations.

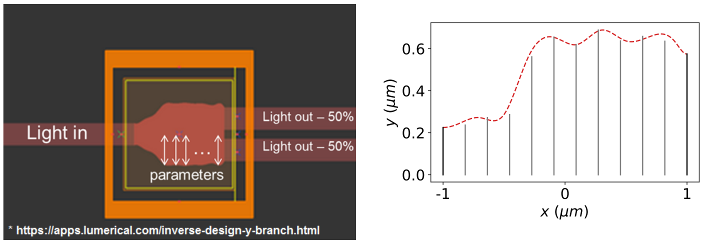


Figure 8.1: Example image of a power splitter (left) and its parametric 10-parameter representation (right).

Among those parameterizations that produced curves of high efficiency (FoM $\geq 97\%$) we observe two principal clusters of curves (Figure 8.2). We observe these curves in Principal Component Analysis (PCA) applied both to the 10-parameter curves and to the scattering coefficients of the interpolated curves (Figure 8.3).

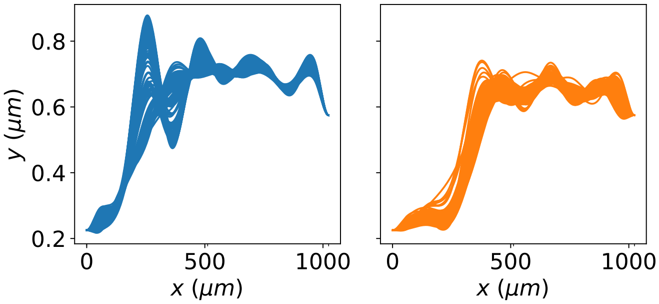


Figure 8.2: Two principal clusters of high-efficiency curves in the provided data.

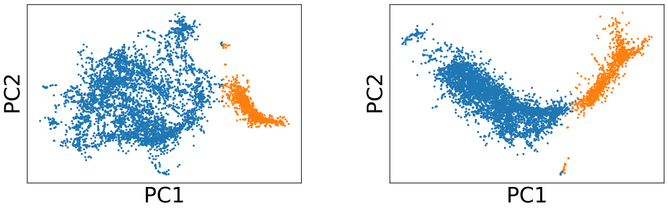


Figure 8.3: Principal components analysis of (left) 10-parameter curves and (right) scattering coefficients of interpolated curves, with a colouring of the principal clusters of high-efficiency curves in the provided data.

Due to the nature of the data (which were generated by an iterative optimization process), we cannot consider the data to be drawn in an *i.i.d.* fashion from the manifold of appropriate solutions. Also the distribution of figures of merit is inherently biased towards 1 since the gradient steps become smaller as the designs become closer to the optimum. Another consideration is that minute differences in efficiency between low-performance splitters are unimportant, while small distinctions in high-performing designs are of great interest. As a result we remove all designs with Figure of Merit smaller than 0.6 and transform the figure-of-merit to a new scale by using the transformation

$$FoM' = -\log(1 - FoM^2).$$

Finally we subsample the data (removing 15% of the total data points) by local density in order to remove highly similar designs from the training data. Figure 8.4 shows the distribution of figures of merit before and after the preprocessing of data.

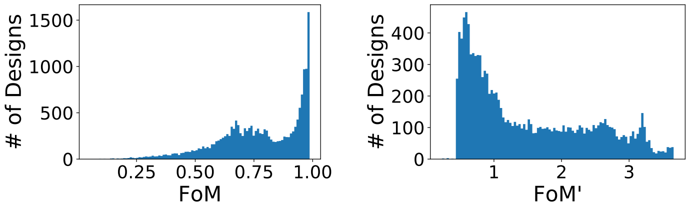


Figure 8.4: Histogram of figures of merit before and after data preprocessing.

8.3 Machine learning approach

We propose to train a *generator*, denoted by G: given a randomly generated vector $Z \in \mathbb{R}^d$, G will output a candidate shape $G(Z)$. It is important that any design can actually be fabricated by practitioners. Hence the generator will be trained to generate shapes that not only have good performance according to the Figure of Merit but are also sufficiently smooth. While the space of all shapes is *a priori* infinite-dimensional, we will assume that the space of *good* shapes, meaning physically possible and with good Figure of Merit, is in fact of relatively low dimension; after G has been successfully trained, it will, with high probability, output shapes in this low-dimensional subspace.

We propose to use a modified version of a generative adversarial network (GAN) to generate the candidate shapes. As in the GAN framework (see Figure 8.5) there are two components:

1. The generator, which will need to take in a latent random vector $Z \in \mathbb{R}^d$ and generate an binary image $x(u)$, where u is in $[0, 1] \times [0, 1]$;
2. A discriminator, which will predict the performance of a device designed from an image $x(u)$.

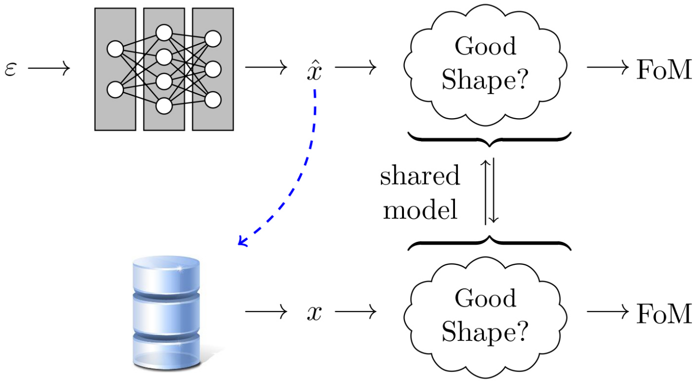


Figure 8.5: Schematic description of a GAN.

This modified GAN will be a generative model that receives reinforcement based on the "performance of circuit" predicted by the surrogate model (see Figure 8.6). In the following sections we provide preliminary approaches (serving as "proof of concept") for both the generation and discrimination tasks, where the latter is considered via scattering transforms and topological data analysis.

8.4 The scattering transform

The scattering transform is a wavelet-based feedforward network that was introduced by S. Mallat[1]: it inputs a signal $f \in \mathbf{L}^2(\mathbb{R}^d)$ and outputs a sequence of numbers that we call "scattering coefficients." These scattering coefficients encode important information about the signal and have been used for machine learning tasks in fields such as audio processing [4], medical signal processing [5], and quantum

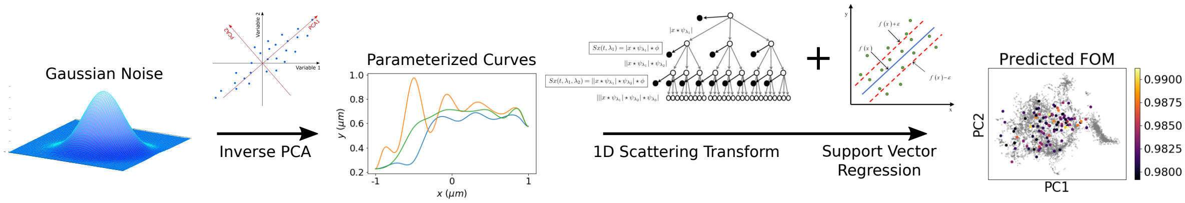

Gaussian Noise
Variable 2
Variable 1
PCA1
PCA2
Inverse PCA
Parameterized Curves
1.0
0.8
0.6
0.4
0.2
-1
0
1
y (μm)
x (μm)
$|x \star \psi_{\lambda_1}|$
$Sx(t,\lambda_1) = |x \star \psi_{\lambda_1}| \star \phi$
$||x \star \psi_{\lambda_1}| \star \psi_{\lambda_2}|$
$Sx(t,\lambda_1,\lambda_2) = ||x \star \psi_{\lambda_1}| \star \psi_{\lambda_2}| \star \phi$
$|||x \star \psi_{\lambda_1}| \star \psi_{\lambda_2}| \star \psi_{\lambda_3}|$
1D Scattering Transform
+
$f(x)+\varepsilon$
$f(x)$
$f(x)-\varepsilon$
y
x
Support Vector Regression
Predicted FOM
PC2
PC1
0.9900
0.9875
0.9850
0.9825
0.9800

Figure 8.6: Schematic description of the proposed experimental approach.

chemistry [6]. The architecture of this network is modelled after a convolutional neural network, using multi-layered alternating cascades of convolutions against filters and pointwise nonlinearities as illustrated in Figure 8.7. Unlike standard CNNs, however, the scattering transform uses pre-designed, wavelet filters rather than filters learned from training data. The use of designed filters allows the user to design a network with invariance and stability properties that may be desirable for a given task.

The scattering transform can be defined formally as follows. Let $J \in \mathbb{Z}$ and let $\psi \in \mathbf{L}^2(\mathbb{R}^d)$ be a fixed "mother-wavelet" (a mean zero function such that $\|\psi\|_2 = 1$). For $j \leq J$ let $\psi_j(x) = \frac{1}{2^j}\psi\left(\frac{x}{2^j}\right)$ and

$$U[j]f = M(f \star \psi_j),$$

where $M : \mathbf{L}^2(\mathbb{R}^d) \to \mathbf{L}^2(\mathbb{R}^d)$ is the modulus operator $Mf = |f|$. The one-step scattering propagator is an operator $U : \mathbf{L}^2(\mathbb{R}^d) \to \ell^2(\mathbf{L}^2(\mathbf{R}^d))$ defined by

$$Uf = \{U[j]f\}_{j \in \mathbb{Z}}. \tag{8.1}$$

Let ϕ_J be a low-pass filter such that $\{\phi_J, \psi_j\}_{j \leq J}$ satisfies a suitable Littlewood-Paley condition (see [1], Equation 2.7), and let A_J be the averaging operator defined by

$$A_J f = f \star \phi_J.$$

For $j_1, \ldots, j_m \leq J$ we let

$$S[j_1, \ldots, j_m] = A_J U_{j_m} U_{j_{m-1}} \ldots U_{j_1} f$$

and define the scattering transform $S : \mathbf{L}^2(\mathbb{R}^d) \to \ell^2(\mathbf{L}^2(\mathbf{R}^d))$ as

$$Sf \coloneqq \{S[j_1, \ldots, j_m]f : m \geq 0, j_1, \ldots, j_m \in \mathbb{Z}\}.$$

We note that a number of variations of the scattering transforms have been developed: for instance they replace the modulus with another nonlinear activation function σ, or replace wavelets with other classes of filters, or replace A_J with another pooling operator (see e.g. [3] and [2]).

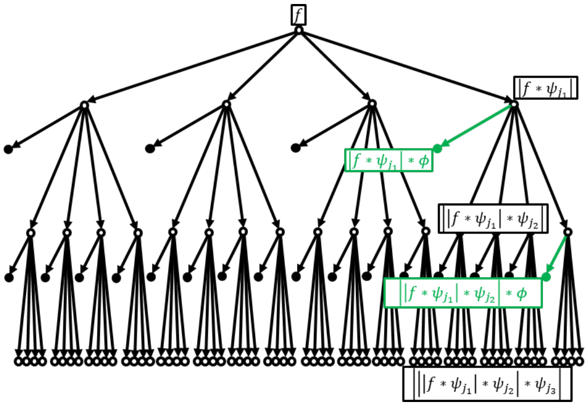

f
$|f \ast \psi_{j_1}|$
$|f \ast \psi_{j_1}| \ast \phi$
$||f \ast \psi_{j_1}| \ast \psi_{j_2}|$
$||f \ast \psi_{j_1}| \ast \psi_{j_2}| \ast \phi$
$|||f \ast \psi_{j_1}| \ast \psi_{j_2}| \ast \psi_{j_3}|$

Figure 8.7: A two-layer scattering network for an input signal f. White dots represent the output of the scattering propagator U and the black dots represent the measurements extracted at each level.

To illustrate how the scattering transform can be applied to scientific problems, we briefly review [3], which used the scattering transform (suitably modified to take the relevant physics into account) to study the formation energies of amorphous Lithium-Silicon (LiSi) systems.

Figure 8.8 displays the first- and second-order wavelet coefficients that result from convolving a LiSi signal f with a cascade of wavelets $\{\psi_\lambda\}_{\lambda\in\Lambda}$. We note that the top left image shows a smearing of the atoms and as the wavelets are dilated the interference pattern spreads to longer-length scales. We observe a similar behaviour for a higher-order wavelet in the second row but with different interference patterns. Since the interference patterns depend upon different distances between the atoms, we say that the wavelet transform separates length scales. In applying a second-order wavelet transform to $|(f*\psi_{j_1})|$ the interference patterns are recombined in a new interference pattern showing the interference within the interference pattern of the prior layer. In this way length scales are recombined.

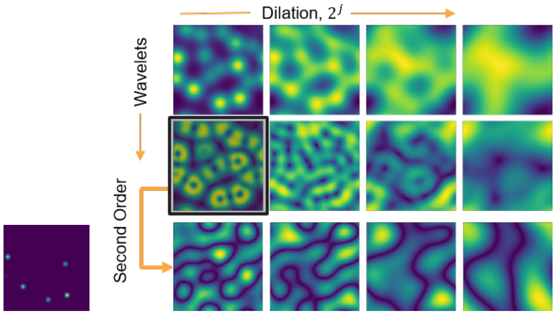


Figure 8.8: Left: A cross-section of a representation of a typical LiSi atomic system. Right: Example wavelet coefficients for the given LiSi system. Higher-order wavelets along the vertical axis of the images and a second-order transform of the indicated system are shown.

Observe that a typical atomic system is composed of locations with an atom and surrounding space and is similar to a nanophotonic image with various components of Si with gaps of air. Hence when we apply the scattering transform to representations of nanophotonic designs, analogous patterns will arise. The wavelets used in [3] were modelled after the solution to the Schrödinger equation for the hydrogen atom. Our current results used default wavelets available on the site [https://www.kymat.io/]. In future work, however, we intend to use a modified class of filters, possibly consisting of shearlets or other commonly used signal processing filters, which are custom-designed for nanophotonics.

For the nanophotonic design problem there are two possible directions with this framework. The first one is to create a scattering network to fit the FoM labels for the 2D images of designs. The second one is to use the wavelet transform coefficients directly to predict the FoM labels. We describe these approaches below.

8.4.1 Regressing over scattering coefficients

The network in this approach is given by training the weights w_0, $w_{\lambda_1 q}$, $w_{\lambda_1\lambda_2 q}$ in the following model.

$$\tilde{y}(f) = w_0 + \sum_{j_1} w_{j_1} \||f*\psi_{j_1}|*\phi\|_1 + \sum_{j_1,j_2} w_{j_1 j_2} \|||f*\psi_{j_1}|*\psi_{j_2}|*\phi\|_1 \tag{8.2}$$

We note that this model does not necessarily have to fit the FoM with high precision: rather an acceptable goal is to distinguish accurately between two designs by preserving the ordering of FoM values. That is, for two designs f_1, f_2 we request that $\tilde{y}(f_1) > \tilde{y}(f_2)$ holds if $\mathrm{FoM}(f_1) > \mathrm{FoM}(f_2)$ holds, or vice versa. Knowledge of how the database is formed can be used to meet this objective. Given a sequence of designs $\{f_{x_i}\}_{i=1}^N$ resulting from the optimization algorithm with f_{x_N} denoting the optimal design, we choose as training data the designs $\{f_{x_{i'}}\}_{i'\in\Theta}$ for $1 \ll i' \ll N$, that is, the designs with large gradient in the FoM with respect to the design parameters.

The performance of the model (8.2) on LiSi systems can be seen in detail in [3]. A possible drawback of this approach is that the wavelet coefficients are summed to a single scattering coefficient for each wavelet, thereby causing a loss of information about the shapes of Si inclusions in the design. This information is encoded in the network through a cascade of scattering coefficients from wavelets at many scales and orders and is recovered at an additional computational cost. This motivates a modification of the model as described below.

8.4.2 Regressing over principal components wavelet coefficients

Rather than regressing over the scattering coefficients we seek to preserve the information encoded in the wavelet coefficients that captures the shape boundaries in the component design. The leftmost wavelet coefficients image of the second row of Figure 8.8 shows how a first-order wavelet transform encodes the perimeters of circular inclusions within the material domain. The second-order wavelet transform of this image shows perimeters of groupings of circular inclusions. This suggests that constituent design shapes at varying scales are effectively captured by the coefficients of the wavelet transform. In other words, given a database with an appropriately diverse set of constituent geometries, the wavelet transform will highlight these shapes. A PCA can then be run on the 2D images of the coefficients (8.1).

A reduced subset of the principal components $\mathcal{B} = \{\phi_i\}_{i=1}^{N}$ with largest singular values may be selected as a basis. The coefficients of the wavelet transform of the design may then be projected onto this basis as follows.

$$\mathbb{P}_{\mathcal{B}}(U[j]f) = \sum_{i=1}^{N_r} \lambda_i \phi_i$$

The components of the projections of scattering coefficients form the features over which we regress in this formulation, rather than the scattering coefficients above. The resulting model may be written as

$$\tilde{y}(f) = w_0 + \sum w_i \lambda_i(f),$$

with λ_i depending on f through the projection. The same procedure may be applied to the second-order wavelet transform coefficients with a PCA basis $\mathcal{B}_1 = \{\phi_{1,i}\}_{i=1}^{N_1}$ over the first-order coefficients and a second basis $\mathcal{B}_2 = \{\phi_{2,i}\}_{i=1}^{N_2}$ over the second-order coefficients. The resulting model is then

$$\tilde{y}(f) = w_0 + \sum w_{1,i}\lambda_{1,i}(f) + \sum w_{2,i}\lambda_{2,i}(f)$$

with $\lambda_{2,i} = \langle U[j_2]U[j_1]f, \phi_{2,i}\rangle$ holding.

8.4.3 Predictive filtering and evaluation of proposed shapes

Initial work shows that 1D scattering coefficients on the cubic spline interpolated curve are sufficient to describe the functionality of the parametric designs. We trained a linear support vector regression on a range of different coordinate systems to find the ideal representation of the data. Figure 8.9 shows the coefficient of determination

$$R^2 = \left(1 - \frac{\sum(y-\hat{y})^2}{\sum(y-\mathbb{E}[y])^2}\right)$$

for each of the following feature vectors: 1. 1D parameterized curve, the 10-dimensional parameter vector; 2. 1D interpolated curve, the high-dimensional vector output of cubic spline interpolated on the 10 parameters; 3. 1D scattering transform, the output of 1D scattering on the interpolated curve; 4. 2D binary image, a matrix representing the cross-sectional view of the device with 1 representing the filled space inside the boundary and 0 representing the space outside; 5. 2D scattering transform, the output of 2D scattering applied to the 2D binary image. We found that in general 1D scattering was sufficient to represent the important features of the device relative to its performance; future work, however, should consider 2D representations as these will allow the analysis of non-parametric devices that are not necessarily symmetric, filled, or described by a low-dimensional parameter vector.

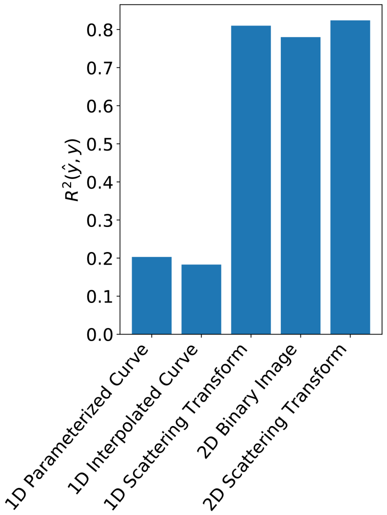


Figure 8.9: Performance of linear support vector regression on different coordinate systems of input data.

Given a good feature vector and surrogate model, we can then generate new data points according to the data distribution and pass these through our regression to find a small subset of designs with high predicted performance. These designs can then be passed through the simulation to determine their true Figure of Merit and thus validate the performance of the model, and can be reincorporated into the training set in the context of active learning. This allows us to refine our predictions and to expand gradually the region of the manifold of good designs from which we are able to sample.

Unfortunately, since the scattering transform is not easily injective, we are not able to generate shapes in the scattering coefficients. Instead we generate data in the PCA space on the parameter vectors. We obtain new parameter vectors by applying inverse PCA, then interpolate these new curves and pass them through the scattering transform in order to predict the Figure of Merit for these new curves. We then filter these curves according to their predicted Figure of Merit, retaining only the top 100 curves. Figure 8.10 shows the distribution of coordinates in the PCA space (on the parameters) of the generated curves relative to high-efficiency curves from the training data (FoM > 0.9) and low-efficiency curves. We see that the generated curves have a distribution that is similar to that of the high-efficiency curves. Also simulating the Figure of Merit for these designs yields overall admissible true figures of merit, with a maximum value of 0.981, just short of the highest overall FoM in the training data of 0.987. Figure 8.11 shows the predicted and actual figures of merit of the generated points and displays these projected onto the PCA space of the training data; we see that these points fill in some gaps on the training data manifold, providing additional data to the model that can be used in active learning to improve our design process.

8.5 Generation of new shapes with kernel classification

In order to find a more general way of generating arbitrary shapes from a small number of parameters, we used kernel classification on a cloud of data in $\mathbb{R}^2$ taken from two different classes. For example, in Figure 8.12a, purple and yellow points represent members of different classes. With the Gaussian kernel we can construct a prediction function that takes a point in $\mathbb{R}^2$ as an input and outputs the probability that the point belongs to the yellow class. For example, if the prediction of one point is greater than 0.5, then we classify it as belonging to the yellow class. We can apply this prediction function to all of the pixels and partition the figure into two parts. The border between the two regions is a curve that we can use to compute the efficiency of the shape.

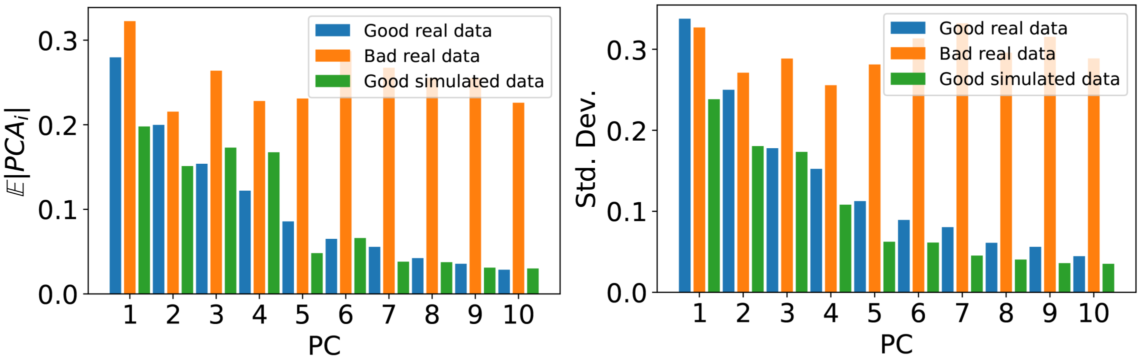


Figure 8.10: Absolute value and standard deviation of principal components of high-quality designs from training data, low-quality simulated designs from training data, and generated designs filtered by the surrogate model.

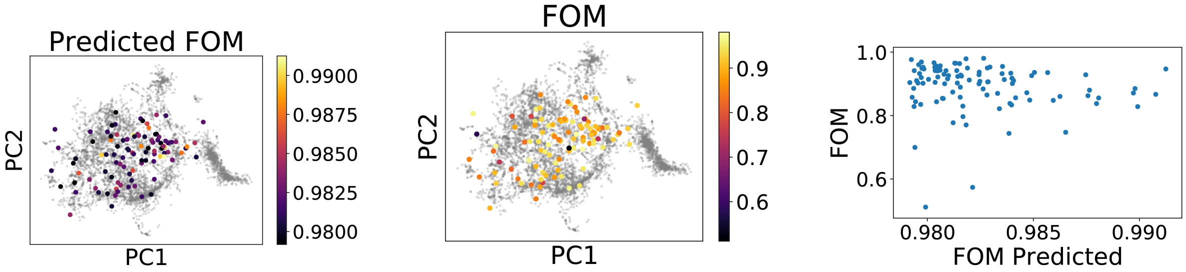
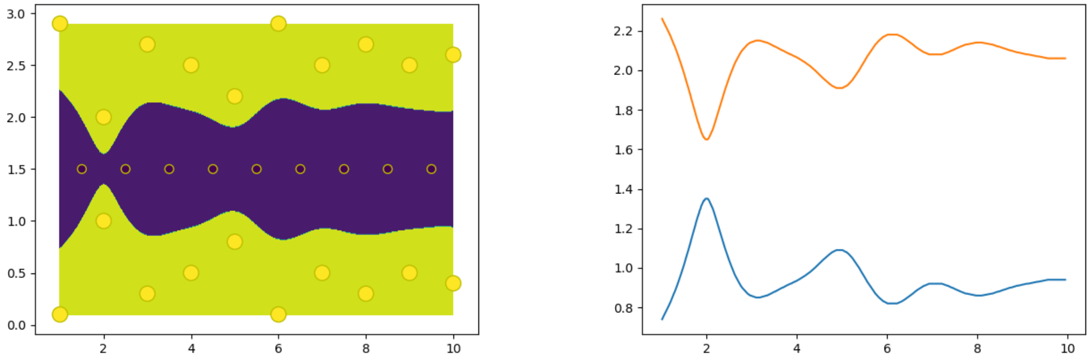


Figure 8.11: Figure of Merit values calculated from photonics simulation of surrogate-model-generated designs compared with values predicted by support vector regression.



(a) A Gaussian kernel classification with symmetric data as an analogue to the construction with cubic spline. Points display the training data and the background illustrates the decision boundary.

(b) The border of decision from the Gaussian kernel classification provides a boundary curve as an analogue to the provided training data.

Figure 8.12: Gaussian kernel classification for generation of a symmetric shape.

Next we considered the case where our shape is not necessarily symmetric. In this case the parameters used were the coordinates of all the points, the kernel to be selected, and the hyperparameter for the classification. Unfortunately the large size of the parameter space caused problems. In future work we will attempt to find a good way to use fewer parameters without completely compromising the additional flexibility obtained by using this more general model. Even if it is possible to use fewer parameters, however, performing optimization in the parameter space will remain a difficult task. One of the possible solutions would be to build a generative model that will produce good shapes automatically.

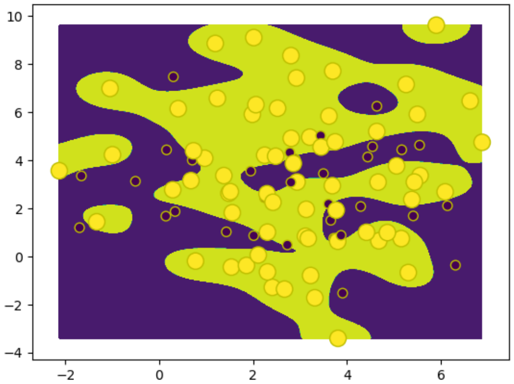
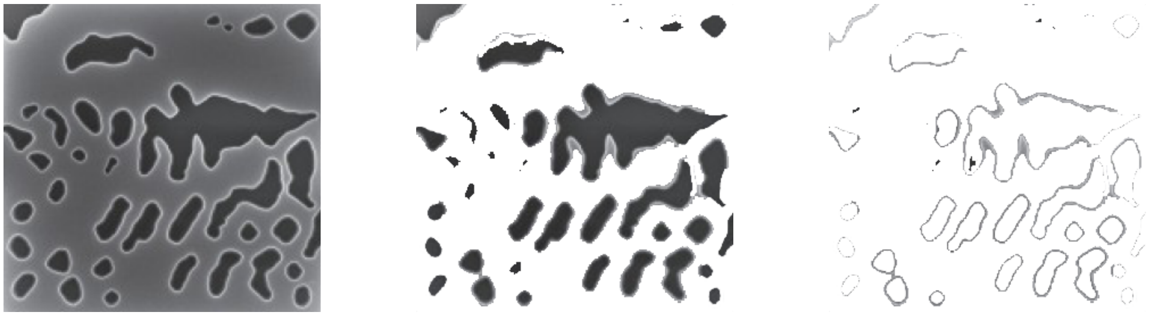


Figure 8.13: A Gaussian kernel classification for a non symmetric data cloud. For this example we use 34 purple points and 66 yellow points generated randomly as training data.

8.6 Classification of data sets by means of homology

Simplicial, cellular, cubical, and other types of homology are the mathematical tools linking together the algebraic and topological properties of a manifold. Ideally we model the dataset from a given experiment as lying on a manifold of a certain dimension. Based on how we model the manifold, we must decide which version of homology is the most appropriate. In analyzing 2D images, which consist of pixels, the cubical homology is a good fit. Images with high resolution contain hundreds of pixels. Homology calculations can take days if carried out by hand while computational tools such as CHomp [10] can do this job in less than one minute. Cubical homology has been comprehensively studied in [7] and reviewed in [8]. CHomp provides homology computations for simplicial, cubical, and relative homology as well as various other computations.

Here we examine the output of CHomp applied to an image of a device design from [9]. Figure 8.14a displays a 2D graphical representation of the device. The image will be treated as a cubical complex and we will study its zeroth- and first-order homology. It is possible, however, to use relative homology to extract more information from the image, which could potentially result in a better classification of the data set.



(a) 2D image representation of inverse-design device from [9].

(b) 2D image of the device with the background of the image removed. The image contains 77 connected components as well as a contour.

(c) 2D image of the device with only shape borders retained. The image contains 97 connected components as well as 16 contours.

Figure 8.14: CHomp homology analysis of a device design.

CHomp produces the zeroth- and first-order homology of the image by treating white pixels as empty space and all other colors as filled space. The number of generators in zeroth-order homology (exponent of $\mathbb{Z}$) stands for the number of connected components in the image. The number of generators for the first-order homology tells us about the number of 2D loops in the image. Since the images are produced in various ways and “purified” (i.e., unwanted pixels are removed), the zeroth- and first-order homology dimensions can have different meanings. We emphasize that the homology loses some information about the geometry of the shape. This method cannot distinguish between a single pixel and a filled disk, or between a circle and an ellipse. The lack of geometric information can be compensated by the study of persistent homology barcodes or the birth/death diagram. Image analysis using homology, together with persistence homology barcodes and birth/death diagrams, can help with the classification of the raw data. Thus we will not have to study each individual piece of data.

Whitening the background image may result in contour creation, which can produce unexpected or undesirable results. This can be observed in Figure 8.14b and Figure 8.14c. This is an artifact of imperfect removal of background or simplification of shape borders. This highlights the importance of carefully preprocessing the input image.

8.7 Applying topological data analysis

We applied topological data analysis [11] to the data to extract topological features. With homology as the only tool it can be hard to compare similar figures with one another. So we add a parameter that increases with time in order to obtain a nested sequence going from the empty set to the complete shape. We then compute the homology of each element within this sequence. This method is called the “persistent homology” method. At any time a topological feature can be born if a new basis of the homology is created and die if a basis disappears or multiples bases merge together. We can build a barcode diagram containing every interval with its birth time and death time. Moreover we can build the persistence diagram, which represents each interval by a point (with the x axis representing the birth time and the y axis the death time). For the implementation of this method we use a Python package called Dionysus 2, which includes tools for topological data analysis.

First we extracted a filtered data set from the data set using a threshold of 0.97 of efficiency. This operation created a data cloud in $\mathbb{R}^{10}$ and we applied a Rips filter to it. This filter creates a ball of radius r centred at a point in the data cloud: we modified the radius over time. We computed the H_0 homology of this filter. We realized that the data points were close to one another and the data set had one main connected component. This gave us a global idea of the data in the parameter space but we could not deduce anything interesting from the barcode diagram.

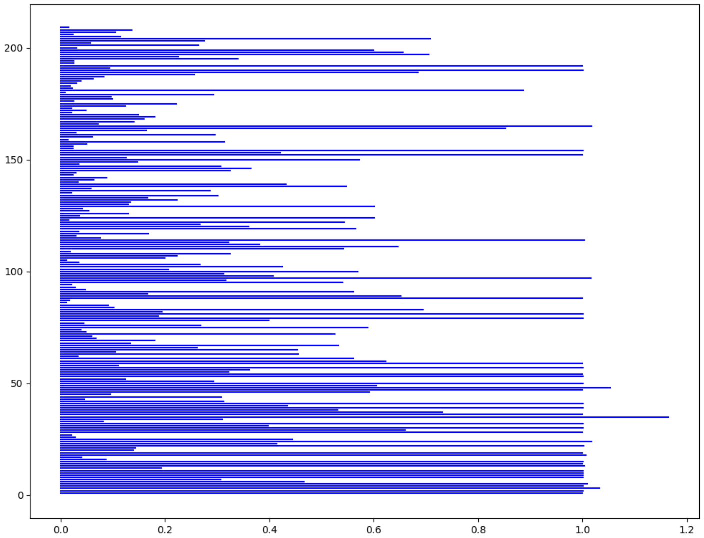


Figure 8.15: Barcode diagram on the parameter space of the dimension 0 homology with a Rips filtration.

In another approach we applied a height filter on the parametric curves from the filtered data set with a threshold of 0.97: we were hoping to find a pattern to be able to construct some clusters. This height filter computes the set of all points above a horizontal line. This line is the parameter of the persistence homology going from the maximum height to the minimum height of the curve. We know that the curves from the data set have only one connected component: thus the filter detects all maxima of the curves. In the barcode diagram the birth represents the first time we encounter the maximum and the length shows how the height of the maximum compares to the two minima in its neighbourhood. We compare all the persistence diagrams using the bottleneck distance, defined as follows.

$$W_\infty(X,Y) = \inf_{\eta : X \to Y} \sup_{x \in X} \|x - \eta(x)\|$$

We were able to find two classes of curves that were of the same form as those found by the other techniques we used.

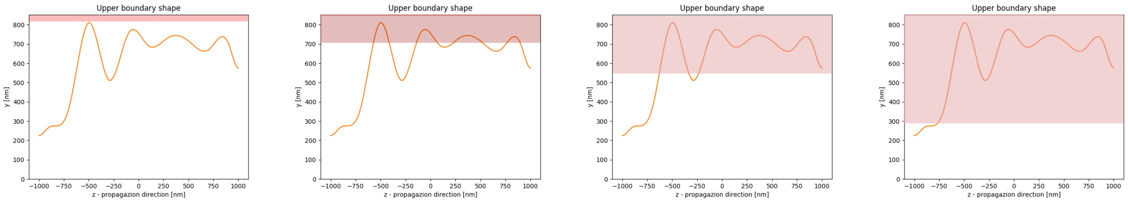


Figure 8.16: Example of a height filter on a curve from the data set.

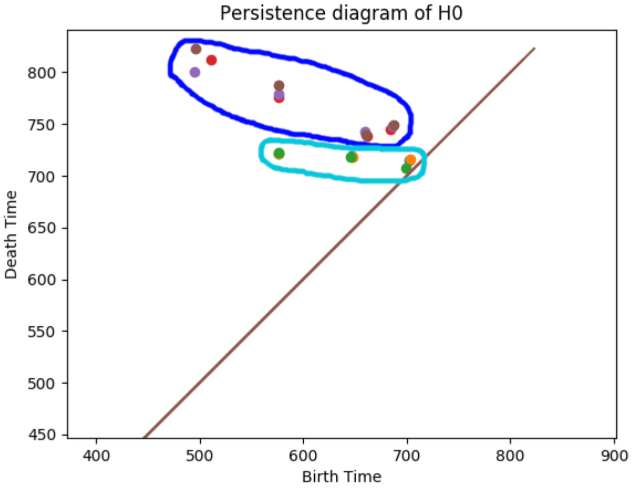
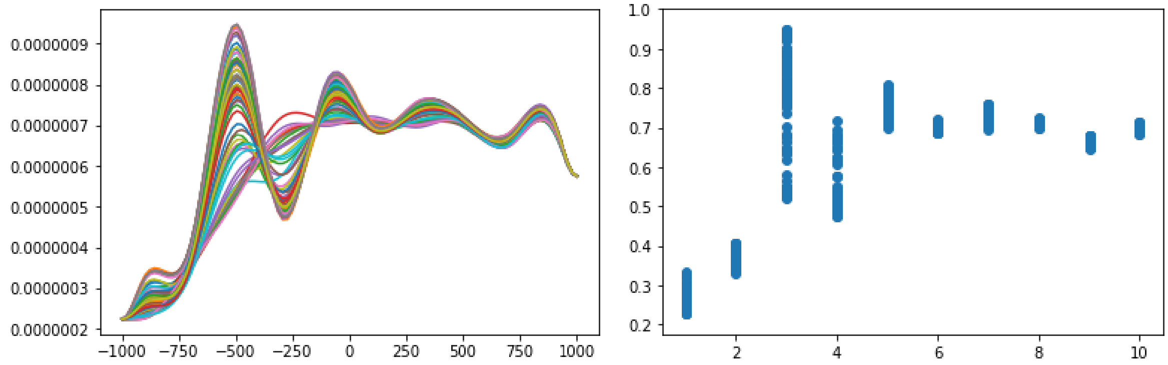


Figure 8.17: Some curves on a persistence diagram, with the two classes of curves.

8.8 A simple data-driven approach for identifying the underlying structure

From a data-driven perspective, a natural way to gain more understanding and obtain (hopefully) more solutions to the original problem is through identifying and extracting patterns from the collection of existing good solutions, then reproducing the patterns to generate other good solutions.

For simplicity, in order to study the best class, we first extract from the existing data set solutions with a Figure of Merit (FoM) greater than 0.985 (Figure 8.18a). Alternatively we can classify solutions using homology. By looking at the scatter plot of the parameters from all the best solutions (Figure 8.18b), we can roughly observe an obvious pattern: there are more degrees of freedom in the third and fourth dimensions than in the other dimensions. Further useful information on the underlying pattern, however, cannot be revealed by that kind of plots, such as the density at different values in each dimension (related to the parameter values).



(a) Filtered Solutions with FoM > 0.985.

(b) Scattered Plot of Filtered Solutions.

Figure 8.18: Plot of solution profiles: the third and fourth dimensions have a larger range of variation than the other dimensions, indicating that they have more degrees of freedom.

To resolve the issue we embed all the scattered data points $\vec{\mathbf{u}}_i$ in $\mathbb{R}^2$, with the ordering indices as the first dimension and the parameter values as the second; then we apply a heat kernel to every one of them and compute the overall heat function F for every $\vec{\mathbf{x}} \in \mathbb{R}^2$ (for some small constant ϵ).

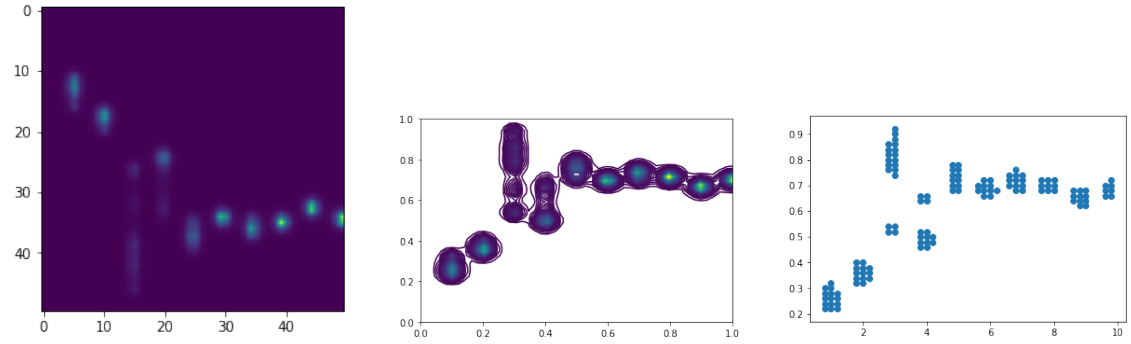
$$F = \sum_{i=1}^{N} \exp\left(-\frac{\|(\vec{\mathbf{x}} - \vec{\mathbf{u}}_i)\|_2}{\epsilon}\right)$$

In this fashion, if a region and its neighbouring area have a higher density or frequency, then it will contain more heat and be easy to identify (Figure 8.19a). Such regions are more noticeable in a contour plot (Figure 8.19b). Moreover, if we impose a threshold on the level surface in order to filter out regions with lower heat, we can recover regions of significance and points in these regions can be interpreted as key nodes to form the best class of solutions (Figure 8.19c).

By investigating the regions of significance or even just the contour plot, we can already draw a conclusion similar to the previous conclusions: the third and fourth dimensions have more degrees of freedom than other dimensions (which are basically restricted to certain values). This information is crucial and useful: if we represent the set of parameters as a real vector $\vec{v}$ in a 10-dimensional space (i.e., $\vec{v} \in \mathbb{R}^{10}$ holds), the parameter sets corresponding to the best solutions are included in a 2-dimensional manifold within $\mathbb{R}^{10}$. Hence if we can determine the expression of this 2-dimensional manifold, we can generate new parameter sets by sampling from this manifold and extrapolate to get new shapes for our devices.

Fortunately we can easily obtain an expression for the manifold in this case. Direct observation reveals that the average of the third and fourth dimensions of all the curves is approximately constant. If we compute this average, we find that it is comprised between 0.56 and 0.71, which is a narrow range because the parameter values range from 0 to values greater than 1. Thus we can apply this restriction along with the individual restrictions on the third and fourth dimensions x_3, x_4 (respectively) to obtain the expression of our underlying manifold.

$$\begin{cases} 0.56 \leq \dfrac{(x_3 + x_4)}{2} \leq 0.7 \\ x_{3_{\min}} < x_3 < x_{3_{\max}} \\ x_{4_{\min}} < x_4 < x_{4_{\max}} \end{cases}$$



(a) Heat Map. **(b) Contour Plot.** **(c) Points with High Heat.**

Figure 8.19: By applying a heat kernel to each parameter point, we can extract significant regions and points. (a) We can roughly identify points in each dimension with high frequencies. (b) This figure displays the contour plot of the heat function. (c) This figure displays a scatter plot of the points with high heat value. Figures (b) and (c) reflect solutions embedded in a two-dimensional manifold.

This system of inequalities corresponds to a “thick” plane segment embedded in the 10-dimensional space. In other more general cases, we can achieve dimension reduction by looking at the variance of each dimension: if the variance of some dimension is lower than a certain threshold, we can treat it as a fixed dimension and neglect it, and then we can deal with the remaining dimensions with tools like manifold learning or topological data analysis (as introduced above).

Continuing our analysis we generate two sets of new solutions (each set containing ten solutions) in slightly different ways: the solutions in the first set are sampled from the points of significance with the above restrictions. For the second set of solutions we only sample x_3, x_4 from the above 2-dimensional manifold and let the value of the parameter in any other dimension be the average of all the values of this parameter for good data. The results displayed in the following table indicate that our approach is reasonable. The high variance in the column titled “Result 1” occurs because we sample points from Figure 8.19c, which introduces degrees of freedom for every dimension. The stability of the values in the column titled “Result 2” validates the statement that the 2-dimensional embedded manifold is actually a “thick” plane segment.

index	Result 1	Result 2
1	0.983	0.9859
2	0.972	0.9852
3	0.972	0.9863
4	0.961	0.9841
5	0.964	0.9862
6	0.984	0.9852
7	0.984	0.9859
8	0.975	0.9863
9	0.964	0.9850
10	0.968	0.9859

8.9 NRC - Short term plans (3 months)

We will implement a set of techniques and methods for generating high-performance candidate devices based on the training data. The developed algorithms are expected to generate free-form shapes for testing and evaluation. We anticipate that some of the techniques will rely on the gradient information of the FoM with respect to the changes in a given shape as a guidance for improvement. Therefore we plan to take the following actions.

- It is necessary to implement libraries that can be used to run Maxwell's equation simulations and extract the gradient information with respect to the changes in the epsilon function (the quantity representing the property of a material). This can be carried out using the adjoint method where only one forward and one backward simulations are needed to extract the gradient information. This will be carried out based on the recent Lumerical inverse design package implementation.
- It will be important to identify whether the optimization and identification of the manifold of good designs must be carried out over a design space in which the epsilon function is allowed to vary continuously, or instead, the epsilon function is discretized, as in a real device scenario. The appropriate way to represent the shapes and parameters will be agreed on and implemented.
- The implementation of the automated way to simulate new shapes and output the results (FoM and the gradients) will be considered to speed up the data collection and training process. Before it is implemented, evaluation of additional designs will be run manually upon demand.

Bibliography

[1] Stéphane Mallat. Group invariant scattering. Communications on Pure and Applied Mathematics, 65(10):1331–1398, 2012.

[2] Wojciech Czaja and Weilin Li. Analysis of time-frequency scattering transforms. Applied and Computational Harmonic Analysis, 2017.

[3] Xavier Brumwell, Paul Sinz, Kwang Jin Kim, Yue Qi, and Matthew Hirn. Steerable wavelet scattering for 3D atomic systems with application to Li-Si energy prediction. In NeurIPS Workshop on Machine Learning for Molecules and Materials, 2018. arXiv:1812.02320.

[4] Joakim Andén and Stéphane Mallat. Multiscale scattering for audio classification. In Proceedings of the ISMIR 2011 conference, pages 657–662, 2011.

[5] V. Chudacek, R. Talmon, J. Anden, S. Mallat, R. R. Coifman, P. Abry, and M. Doret. Low dimensional manifold embedding for scattering coefficients of intrapartum fetal heart rate variability. In 2014 Internat. IEEE Conf. in Medicine and Biology, 2014.

[6] Matthew Hirn, Stéphane Mallat, and Nicolas Poilvert. Wavelet scattering regression of quantum chemical energies. Multiscale Modeling and Simulation, 15(2):827–863, 2017. arXiv:1605.04654.

[7] M. Mrozek, T. Kaczynski, M. Mischaikow. Computational Homology. Springer, 2003.

[8] Pawel Pilarczyk, Pedro Real. Computation of cubical homology, cohomology, and (co)homological operations via chain contraction. Springer Science+Business Media, 41(2):8253–275, 2014.

[9] Alexander Y Piggott, Jesse Lu, Konstantinos G Lagoudakis, Jan Petykiewicz, Thomas M Babinec, and Jelena Vučković. Inverse design and demonstration of a compact and broadband on-chip wavelength demultiplexer. Nature Photonics, 9(6):374, 2015.

[10] Konstantin Mischaikow, Hiroshi Kokubu, Marian Mrozek, Pawel Pilarczyk, Tomas Gedeon, Jean-Philippe Lessard, and Marcio Gameiro. Chomp: Computational homology project. Software available at http://chomp. rutgers. edu, 2014.

[11] Robert Ghrist. Barcodes: The persistent topology of data. Bull. Amer. Math. Soc., 45(1):61–75, 2008.

9 Towards the unsupervised learning of novel RFI sources

Sean Bohun [a]
Rory Coles [a]
Nicholas Bruce [b]
Chris Budd [c]
Ryan Campbell [d]
Seth Siegel [d]
Dave Del Rizzo [e]
Stephen Harrison [e]

[a] University of Ontario Institute of Technology, Oshawa (Ontario) Canada

[b] University of Victoria, Victoria (British Columbia), Canada

[c] University of Bath, Bath, United Kingdom

[d] McGill University, Montréal (Québec), Canada

[e] Dominion Radio Astrophysical Observatory, Kaleden (British Columbia), Canada

October 2020
Les Cahiers du GERAD

Abstract: *We construct a simple algorithm for the separation of different types of short radio signal interference.*

9.1 Introduction

The Dominion Radio Astrophysical Observatory (DRAO) is located in a geographically isolated region near Penticton B.C. This is the site of a multitude of sensitive RF detectors that require a quiet RF spectrum to operate effectively. On any given day the site can experience any number of transient RF signals (typically around 1GHz in frequency) emitted by a variety of sources. Some of these are random noise but the majority have some sort of "organized" human origin. An example of such might be the electronic key used to unlock a car. The human-made signals tend to have distinctive modulation patterns and (digital) signatures. DRAO would like to develop the following capabilities.

- Classify and Cluster the set of known RF sources as they are determined.
- Identify any novel RF sources that have not been previously classified.
- Provide a set of descriptors for each novel source.
- Update the clusters dynamically as novel sources are identified.

After this the hope is that any *novel sources* can be identified with this technique and then eliminated.

One method for doing this might be to use a straight machine learning approach in which an auto-encoder is trained on a large number of labelled signals, which can then be classified. Whilst this is promising in theory it takes time to code up and to train the auto-encoder successfully, and the complexity of the signals means that a lot of data and significant computing power would be needed to do this. The IPSW was too short to implement a full machine learning approach to this problem, although it is a perfectly feasible way to proceed.

Instead the procedure adopted in the IPSW was to make a careful visual inspection of the various signals using both the time series and the spectrogram. The visual inspection quickly revealed a clear pattern of qualitatively different types of signal that could be used for subsequent classification. This was mainly based upon the modulation pattern of the different signals. Two classifiers for the signals were then considered: one was the use of higher order cumulants (essentially moments of the signals). The second was a simple count of the number of peaks of the signal in the time and frequency domains. It was found that a combination of these two classifiers was sufficient to partition the observed signals into classes that agreed with the visual comparison of the signals.

9.2 Background

The location of the DRAO is shown in the figure below.

Radio interference at this location can be detected and measured both as a time series and as a frequency spectrogram. Typically the interference occurs at frequencies of around 400MHz-2GHz and is a mixture of natural and human-made signals. Such signals are usually localized in both time and frequency. This allows separate signals to be identified and studied. Different signals display a characteristic shape in both the time and frequency domains, largely associated with their modulation type. The procedure we adopted in the IPSW study was as follows.

1. Separate the signals using bounding boxes.
2. Produce an indexed dictionary of the different signals.
3. Visually inspect the different signals in both the time and frequency domains.

4. Visually label the different types of observed signals according to their qualitative form.
5. Determine various quantitative classifiers of each signal based on the qualitative inspection. above
6. Partition the different signals using the classifiers.
7. Compare the partition determined in 6 with the labels determined in 4 to assess the reliability of the classifiers.

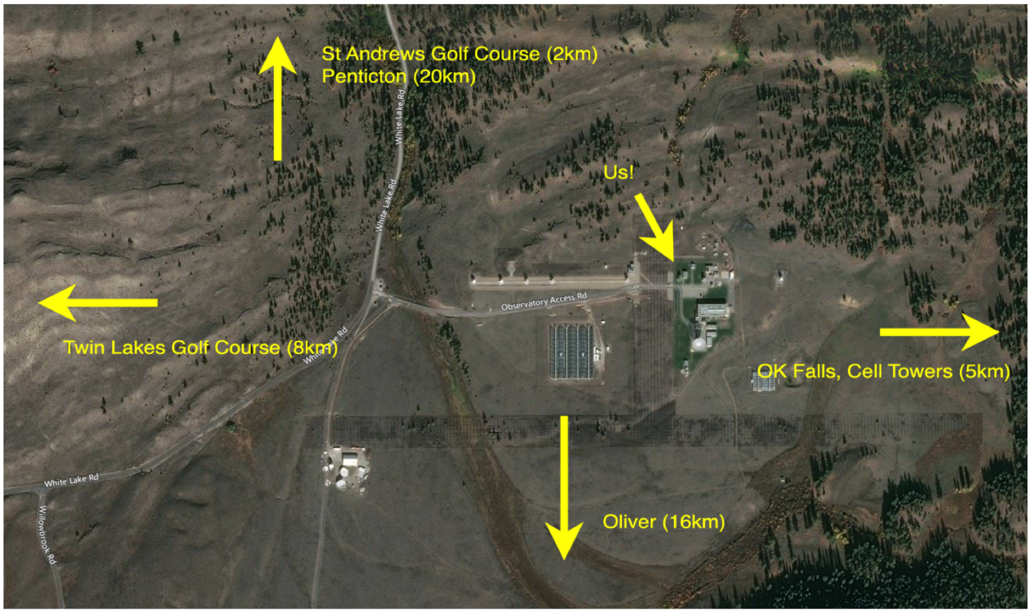

St Andrews Golf Course (2km)
Penticton (20km)
Us!
Twin Lakes Golf Course (8km)
OK Falls, Cell Towers (5km)
Oliver (16km)
White Lake Rd
Observatory Access Rd
Willowbrook Rd

Figure 9.1: The locations of the radio observatory and possible radio sources.

This procedure was completed successfully during the course of the IPSW. It was greatly aided by the work of DRAO members, who had already carried out tasks 1 and 2 before the meeting.

Note that this procedure differs from the usual machine learning approach in which the auto-encoder would be trained on the labelled data and would essentially determine its own classifiers in an automatic fashion. It would be a very good idea to compare this approach with the results of the IPSW but this will take time and computational effort.

9.3 Signal separation

The figure below shows a spectrogram of the measured signals over a frequency range of 435MHz-485MHz for a period of about 145s. In this figure we can see a number of signals at distinct frequencies that are almost certainly due to organized human activity. In contrast the broadband signal at time 110s is probably a short burst of noise.

The signals isolated in time and frequency were separated using an algorithm developed by DRAO, which used a machine learning approach to place a bounding box around them in the spectrogram (as illustrated below). The result was a series of separated signals, which could then be compared both qualitatively and quantitatively. A typical result is shown in the second figure below.

9.4 Qualitative labelling of the different signals

The indexed signals were then examined, both in the time domain (plotted left below) and the spectrogram (plotted right below). The signals have a characteristic form showing different types of modulation. The modulation type allows a rough classification, which we now describe.

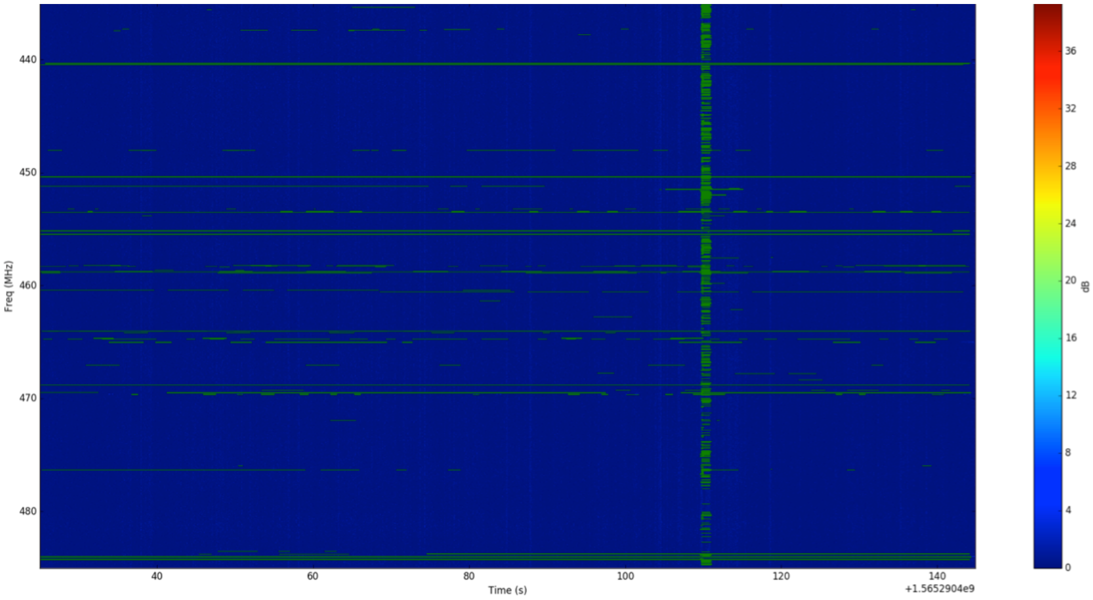


Figure 9.2: A spectrogram of the observed signals over a 145s window.

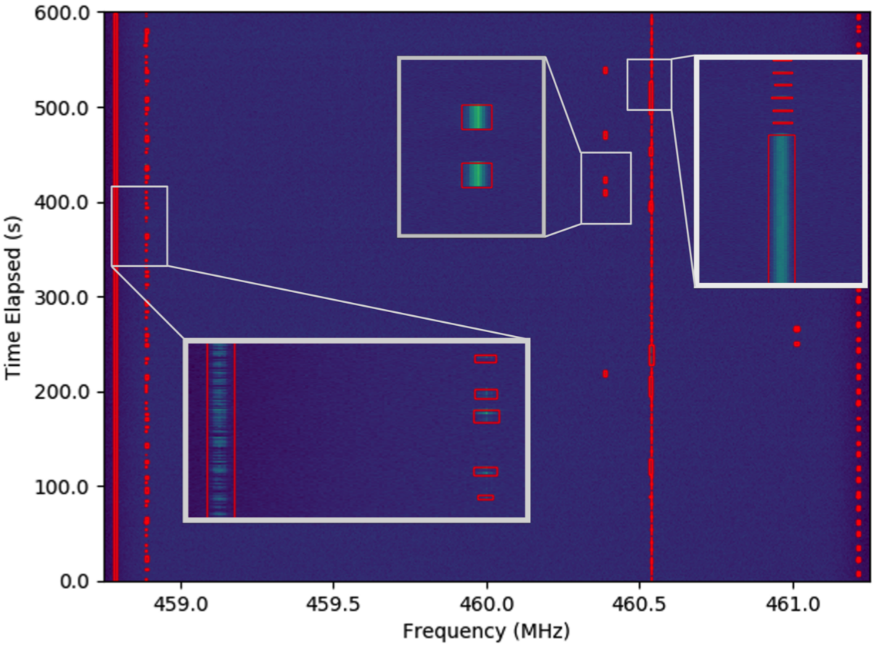


Figure 9.3: Placing boxes (through machine learning) around the signals in the spectrogram.

Case 1: FM This is plotted in Figure 9.5 and is a *short tone burst*. This is a short pulse of *constant amplitude*. The spectrogram shows a main carrier frequency and a series of side bands. This is a characteristic signature of a frequency/phase modulated signal.

Case 2: FSK This is illustrated in Figure 9.6. In this figure we see a signal with a series of pulses of amplitudes 1:2. The spectrogram shows evidence of a limited number of frequency components. We suspect that this is evidence of a *frequency shift key* (FSK) modulated signal.

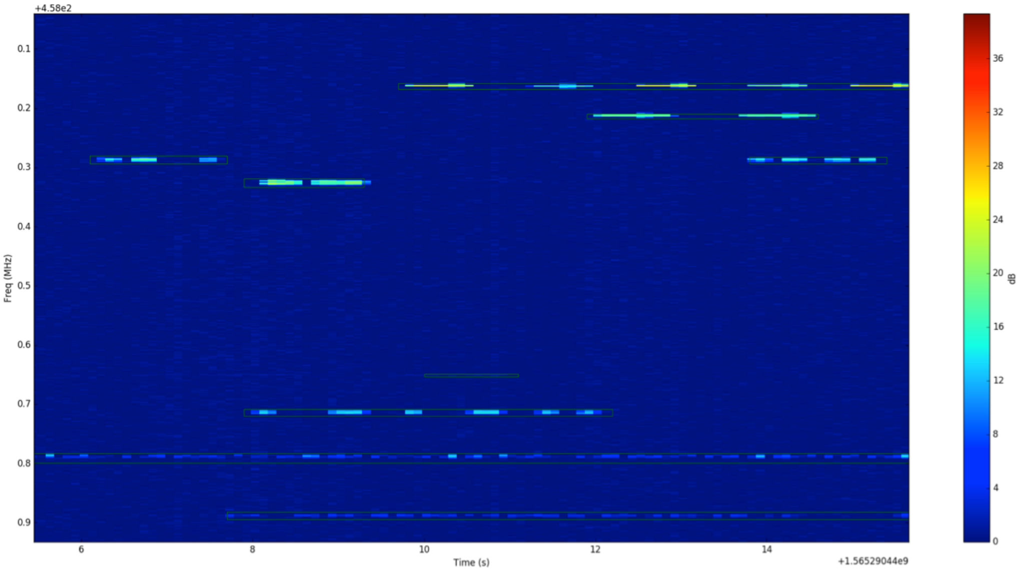


Figure 9.4: The separated signals in the spectrogram.

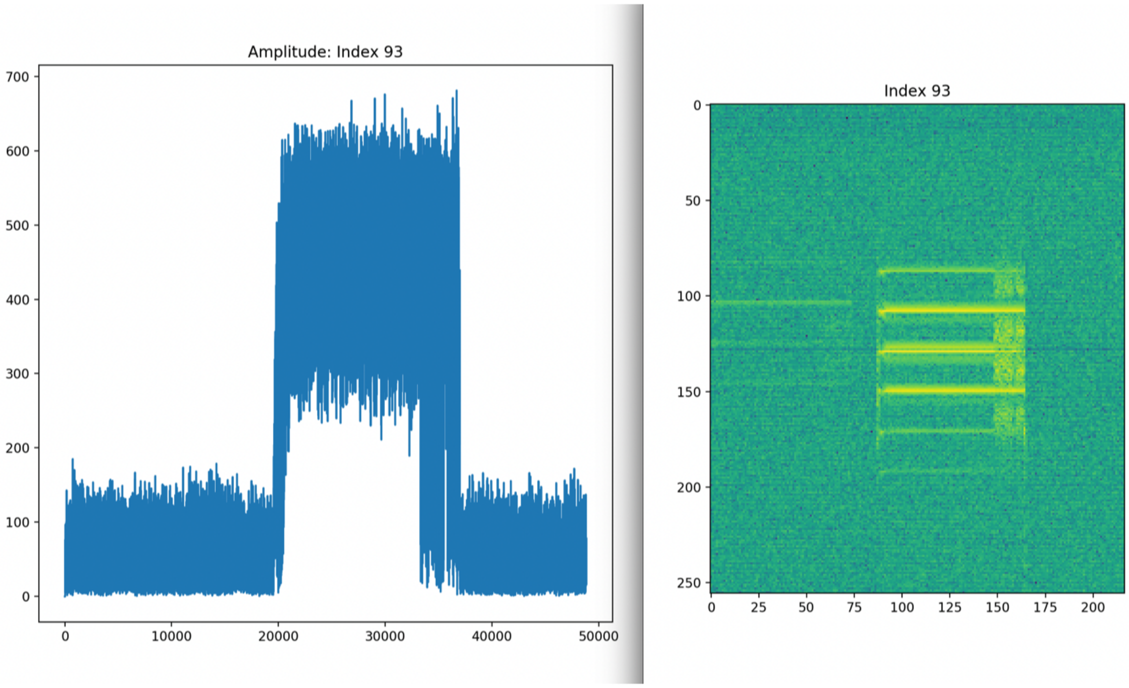


Figure 9.5: A short FM tone burst.

Case 3: ASK The next signal (which is very similar to the one before and is illustrated in Figure 9.7) appears as a series of short pulses of amplitudes in proportions 1:2. The frequency of these pulses shows significantly more components than the previous signal. There is evidence here of *Amplitude Shift Key* (ASK) modulation. The signature of this signal is consistent with a key fob used to unlock a car.

Case 4: Noisy burst The signal illustrated in Figure 9.8 shows a single pulse with a broadband spectrum. This shows no evidence of any form of modulation.

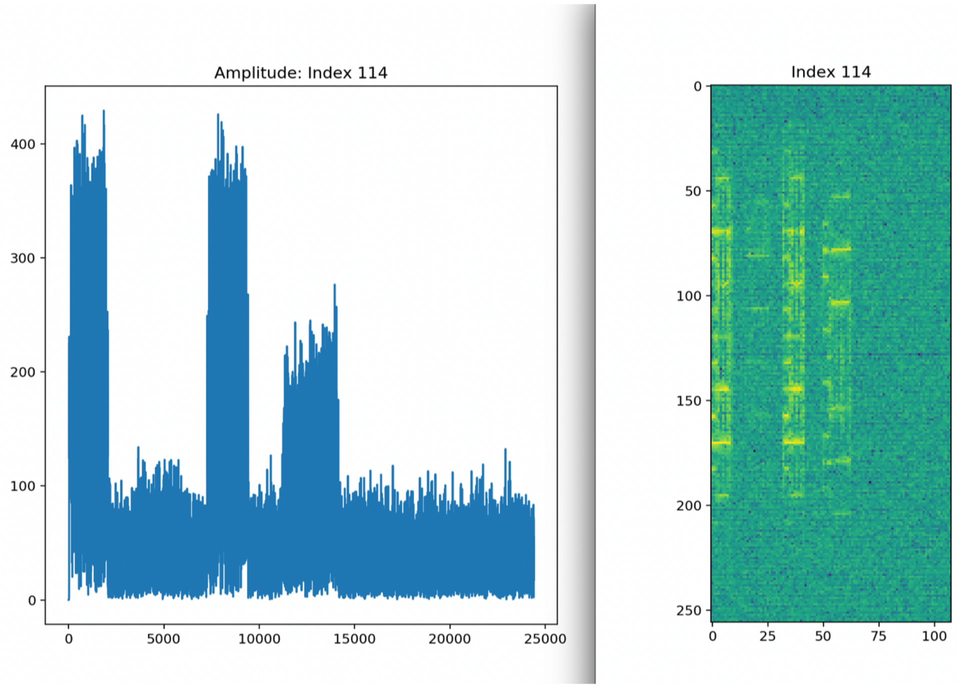


Figure 9.6: A FSK modulated signal.

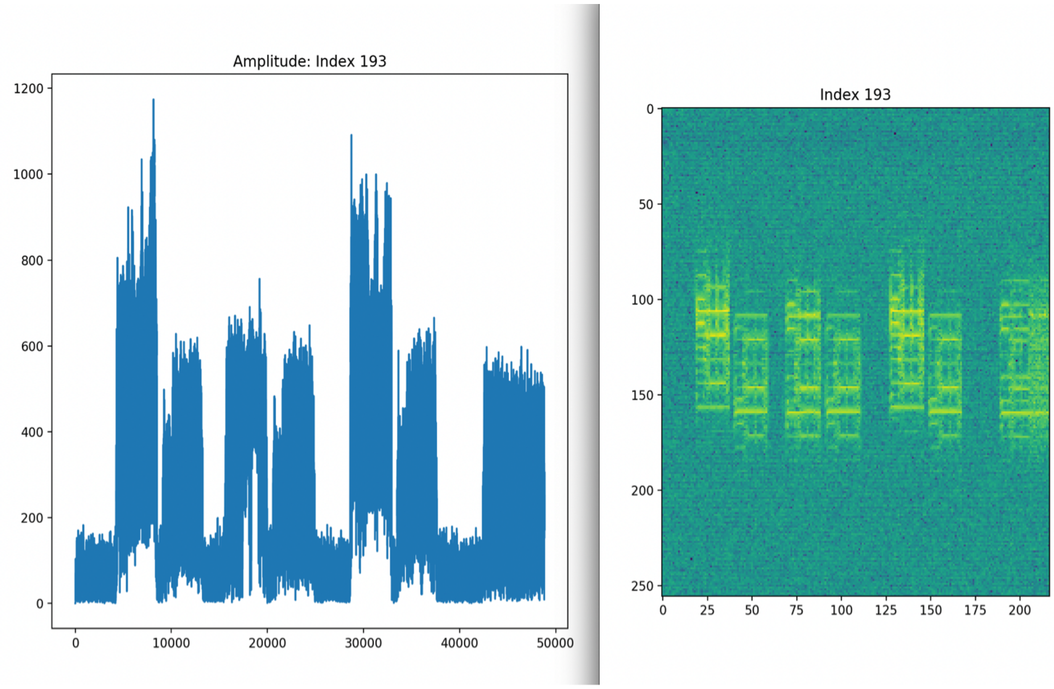


Figure 9.7: Amplitude Shift Key (ASK).

Case 5: Unknown box The signal illustrated in Figure 9.9 shows a single (long) pulse with a broadband spectrum superimposed on a broader band noisy burst. This shows no evidence of any form of modulation. We call this a box given its characteristic spectrogram signature.

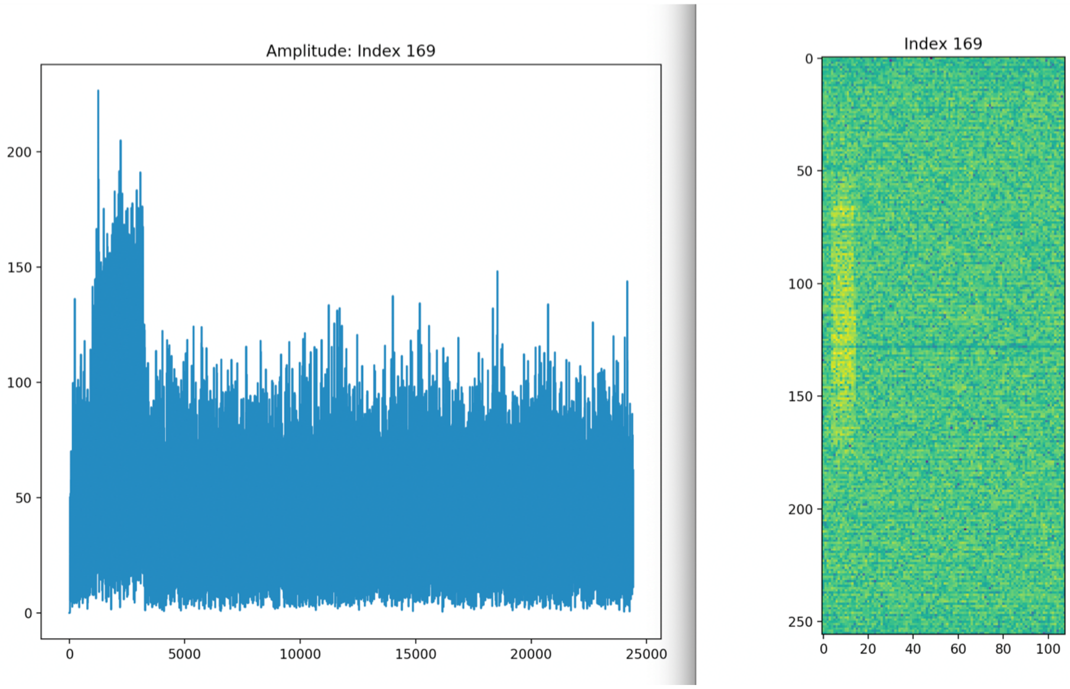


Figure 9.8: A noisy burst.

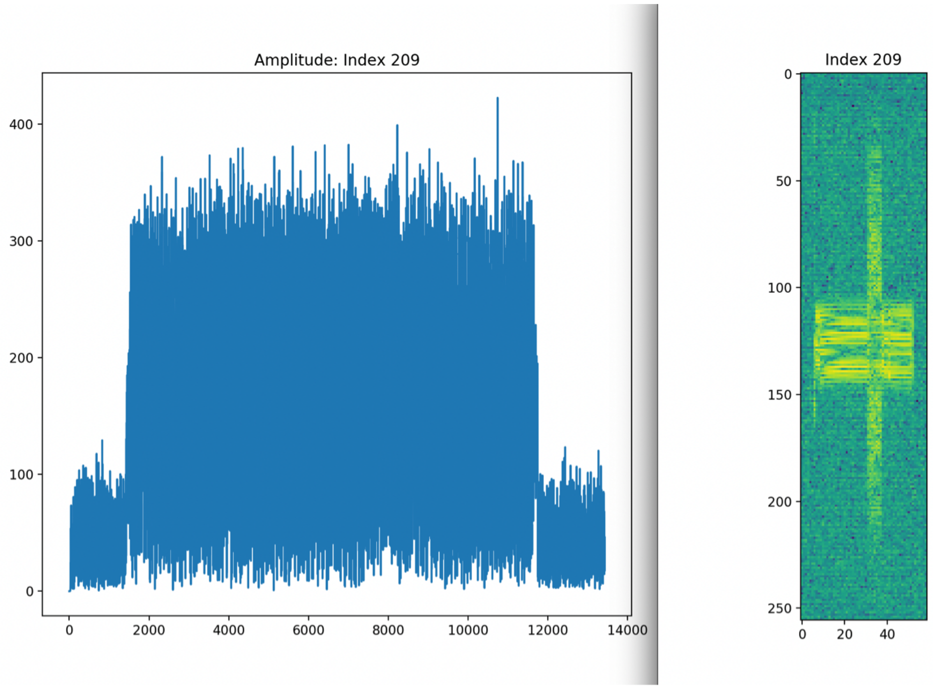


Figure 9.9: A “box” signal.

9.5 Differentiating between different signals

We now consider more quantitative ways of discriminating between types of signal.

9.5.1 Signal features

Signals clearly differ in the time of day when they are sent, their length, and the frequency (or frequencies) at which they are sent. These all give crude separation measures and may be correlated with different types of human behaviour such as leaving for work at a certain time. Signals also have different modulation types for their information content. These can be classified as *Analogue Modulation* types: AM, SSB, DSB, FM, PM, and *Digital Modulation* types: ASK, FSK, PSK, QAM, BPSK. These act as more precise separation measures as it is likely that different devices will employ different forms of modulation.

It is natural to consider separating signals by looking at their modulation types. A well established way, which is claimed to do this, is the method of using *higher order cumulants* [1]. Suppose that the received signal (assumed to be a complex vector) is $\mathbf{u}(t)$. We can define the *higher-order moment* M_{pq} as

$$M_{pq} = E\left[\mathbf{u}^{p-q}\left(\mathbf{u}^*\right)^q\right]. \tag{9.1}$$

Two of the higher-order cumulants are then defined as:

$$C_{42} = M_{42} - |M_{20}|^2 - 2\, M_{21}^2 \tag{9.2}$$

and

$$C_{63} = M_{63} - 9\, M_{21}\, M_{42} + 12\, M_{21}^3 - 3\, M_{20}\, M_{43} - 3\, M_{22}\, M_{41} + 18\, M_{20}\, M_{21}\, M_{22}. \tag{9.3}$$

It is claimed in the literature that these two cumulants are effective in distinguishing between different modulation types. We remain unconvinced about this, however. The articles base these claims on the use of polynomial supervised learning based on cumulants obtained from clean synthetic data. In our case we are using unsupervised learning from real, noisy, data. In this case the cumulants are much less effective and need to be augmented with other measures of the signal. Various features were considered, in particular the following 13.

1. Center Frequency (Hz)
2. Bandwidth (Hz)
3. C_{42} (4th order power cumulant)
4. C_{63} (6th order power cumulant)
5. Transmission length (in seconds)
6. P_{db} (2nd order power cumulant)
7. t_{pk} (number of peaks in the time direction)
8. f_{pk} (number of peaks in the frequency direction)
9. Prominence of the major frequency peak
10. Average spacing of the frequency peaks
11. Average spacing of the time peaks
12. Normalized power centroid in the time direction
13. Width in the frequency direction (channels)

In this list the features 3 and 4 are the cumulants described above. The features 7 and 8 are new ones that were identified as important during the IPSW. The motivation for using these is that many of the signals observed showed distinct time and frequency peaks as a result of the modulation used. The number of these peaks could then be determined directly from the spectrogram/FFT. An example of this can be seen in Figure 9.10 in which we see a signal with one time peak and 5 clear frequency peaks.

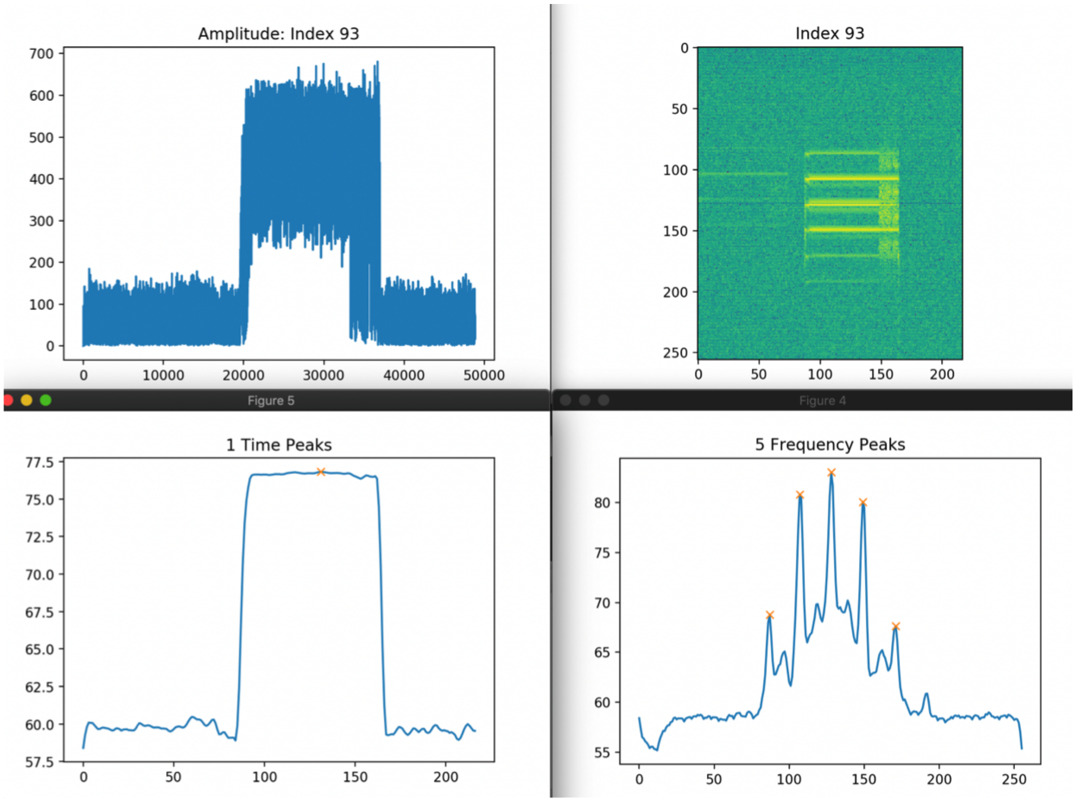


Figure 9.10: Identification of the number of time (t_{pk}) and frequency (f_{pk}) peaks for signal index 93.

9.5.2 Signal classification

Using the features above we could allocate to each signal a 4D vector classifier

$$C = (C_{42}, C_{63}, t_{pk}, f_{pk}).$$

The features were then divided into clusters using the *dbscvan* (density-based spatial clustering of applications with noise) algorithm in Matlab. According to the Matlab documentation the routine `idx = dbscan(X, epsilon, minpts)` partitions observations stored in the n-by-p data matrix X into clusters using the DBSCAN algorithm. DBSCAN [2] clusters the observations (or points) based on a neighbourhood search with radius epsilon and a minimum number of neighbours (minpts) required to identify a core point. The function returns an n-by-1 vector (idx) containing a cluster index for each observation.

In order to compare the effectiveness of the different classifiers we decided to look at the effects of two separate clusterings. The first one is based on the identifiers (C_{42} and C_{63}) and the second one on the identifiers (t_{pk} and f_{pk}). In each case the algorithm was instructed to divide the signals into four clusters. The results are displayed in Figures 9.11 and 9.12 with the clusters coloured appropriately. Note the parabolic form (close to the origin) of the cluster of the cumulants in Figure 9.11. This is evidence of these signals (coloured blue) being Gaussian noise. The results in Figure 9.12 show a reasonable separation between different signal types.

We then give each signal a 2D classifier

$$I = (cc, tf),$$

which is the combined classifier index from the cumulants and the time-frequency analysis taken separately. These two indices appear to be independent and give different information about the signal. Evidence for this is given by plotting one against the other as can be seen in Figure 9.13.

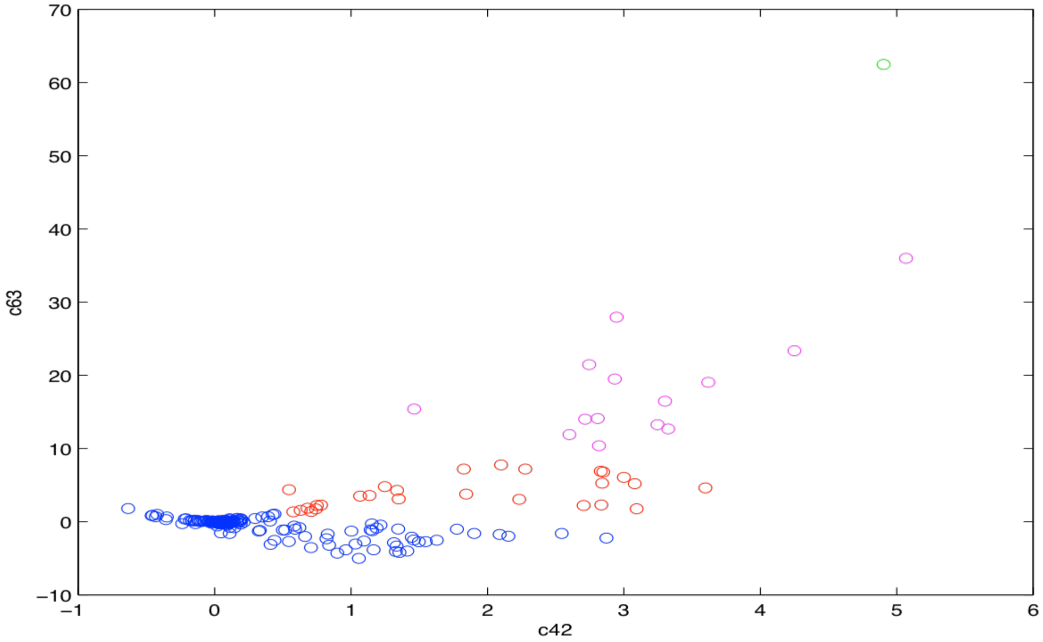


Figure 9.11: Clustering based on the higher-order cumulants.

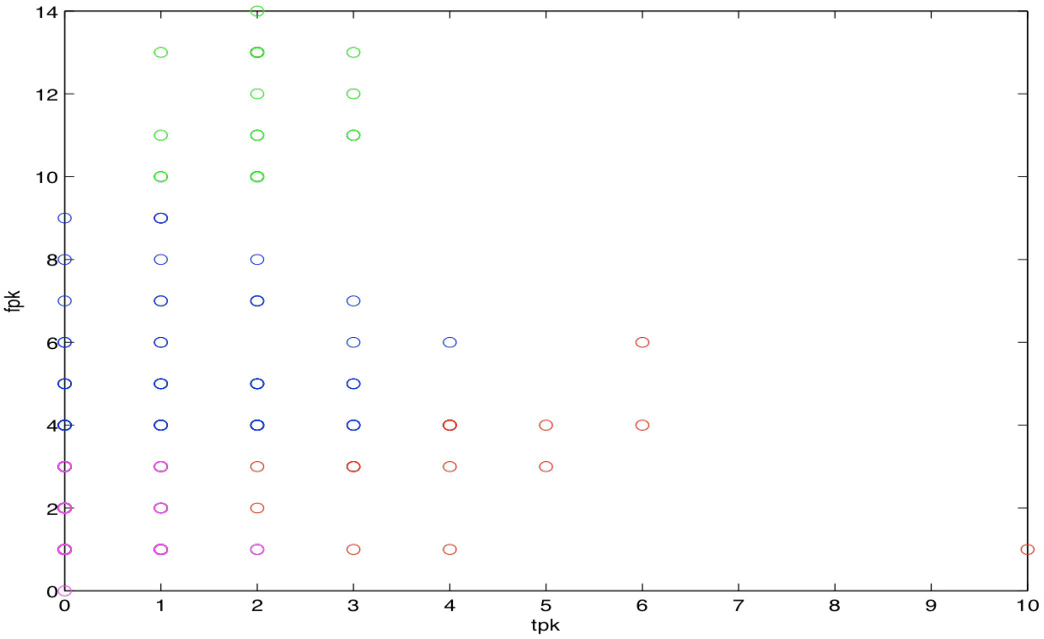


Figure 9.12: Clustering based on the time and frequency peaks.

From our study of the actual data the index I appears to provide a useful identification of the different types of signal and allows new signals to be identified. This is a simple form of hierarchical clustering.

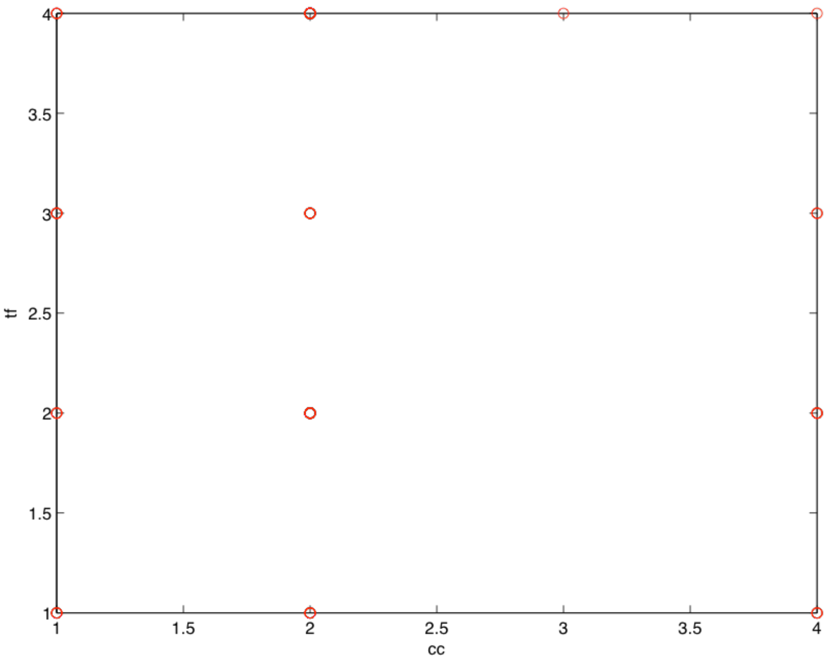


Figure 9.13: A comparison of the cumulant classifier with the time-frequency classifier, showing that these two indices are independent.

9.5.3 The effectiveness of the clustering index I

Using the clustering index I we divided the signals into a number of different clusters. These are presented in Figures 9.14, 9.15, 9.16, 9.17, and 9.18 below. A visual inspection shows that the index is doing a good job in both grouping similar signals and separating different signals.

9.6 An auto-encoder approach

Our work during the IPSW also included a direct attempt to classify the signals using an auto-encoder. Two methods were tried separately using a convolutional auto-encoder on the baseband time-series data:

1. The first idea was to present all the signals to the encoder and use the latent feature space for clustering;
2. The second idea is to avoid clustering and instead train the auto-encoder on the full dataset and then compute the difference between the input and the reconstruction. If the error is large enough it is “anomalous.”

In the second case the auto-encoder was trained on the whole dataset with a power P_{db} index greater than 40 dBm, then compared with simulated pure white noise. Next a collection of real signals known to be noisy was used for training and the rest were used for testing.

The results of this particular piece of work were inconclusive during the workshop but this was mainly due to time limitations. A full machine learning approach is likely to be more successful if one has more time to train the network and more careful classifiers.

Figures 9.20, 9.21, and 9.22 show the process of separating those signals using a parallel coordinates plot.

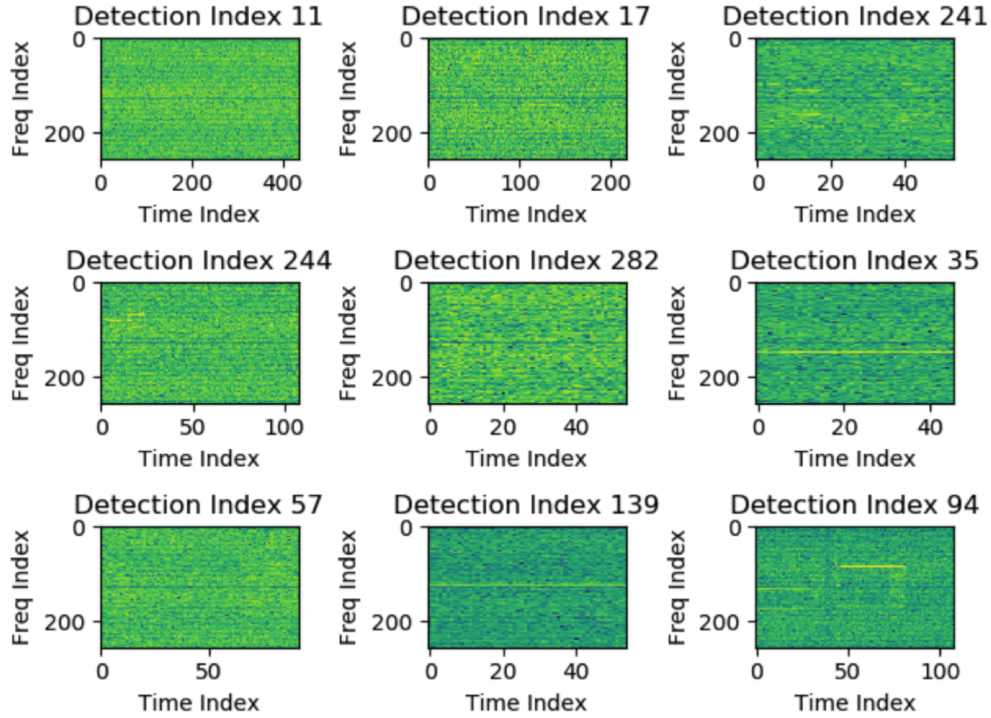


Figure 9.14: Cluster 1: $I = (2, 4)$**.**

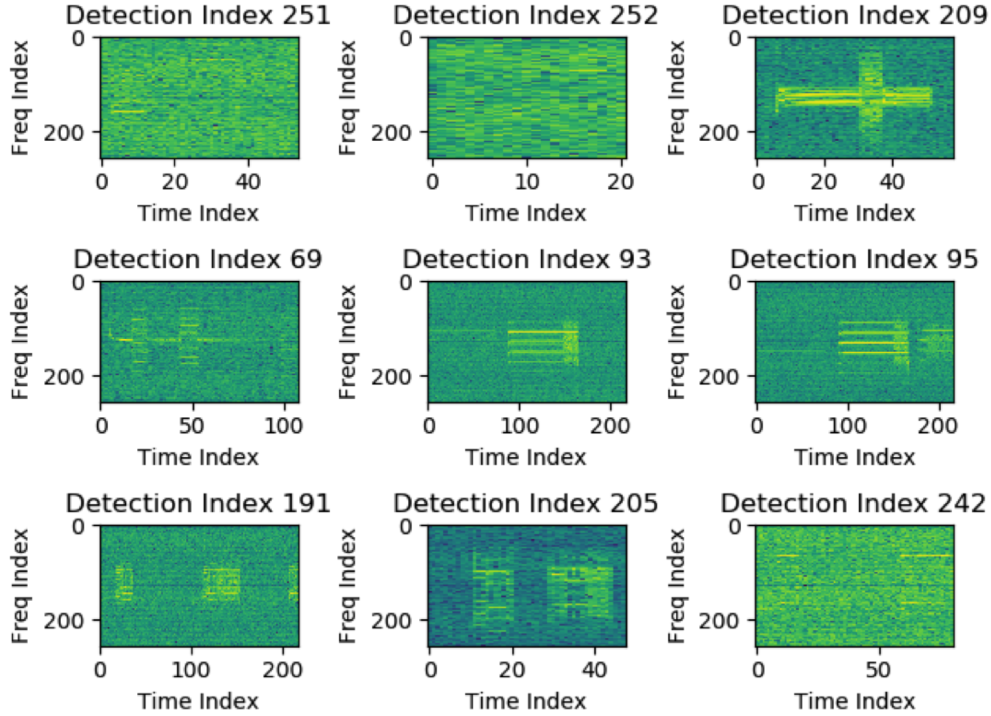


Figure 9.15: Cluster 2: $I = (2, 2)$**.**

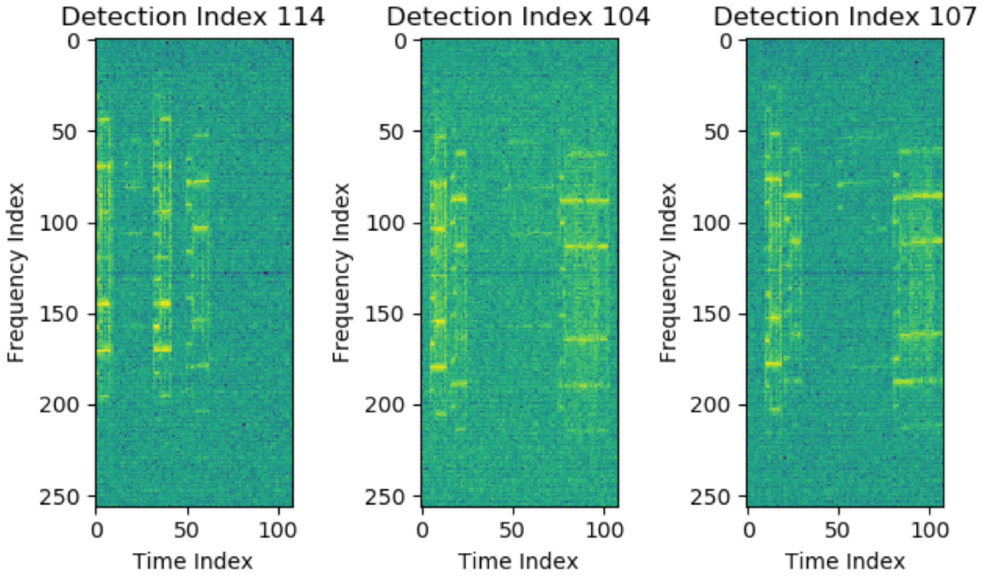


Figure 9.16: Cluster 3: $I = (1, 3)$.

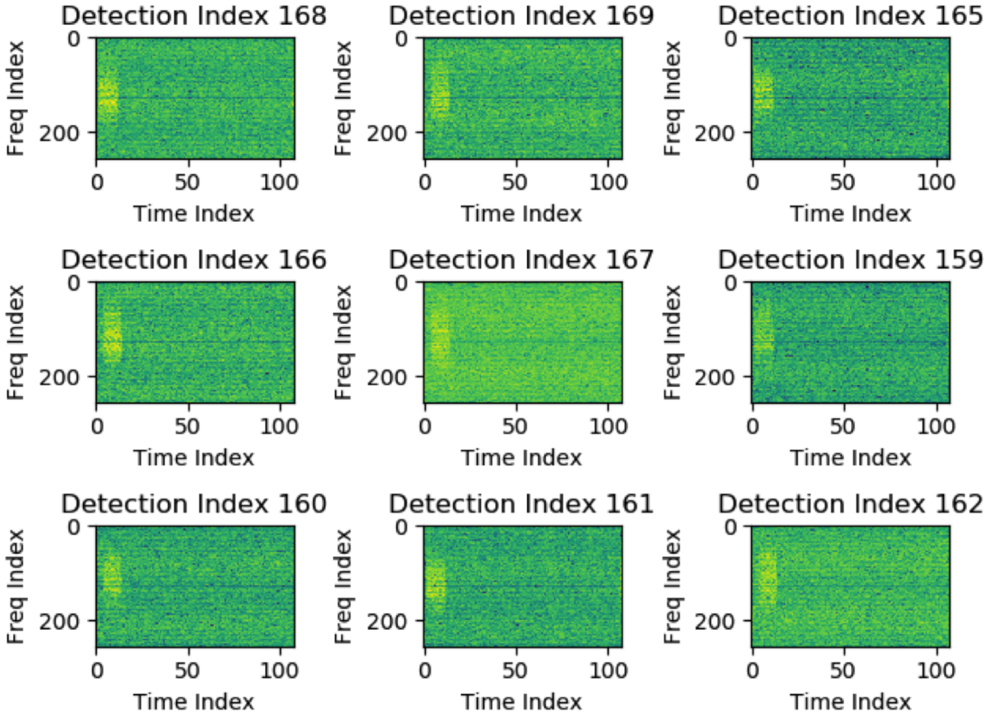


Figure 9.17: Cluster 4: $I = (1, 4)$.

9.7 Conclusions

Building on the unsupervised detection, it was found that cumulants C_{42} and C_{63} (when taken alone) were insufficient to cluster incoming RFI signals. This is in contrast to other researchers' claims that cumulants are sufficient. This claim, however, was based on supervised learning algorithms using synthetic data and not unsupervised learning on real data.

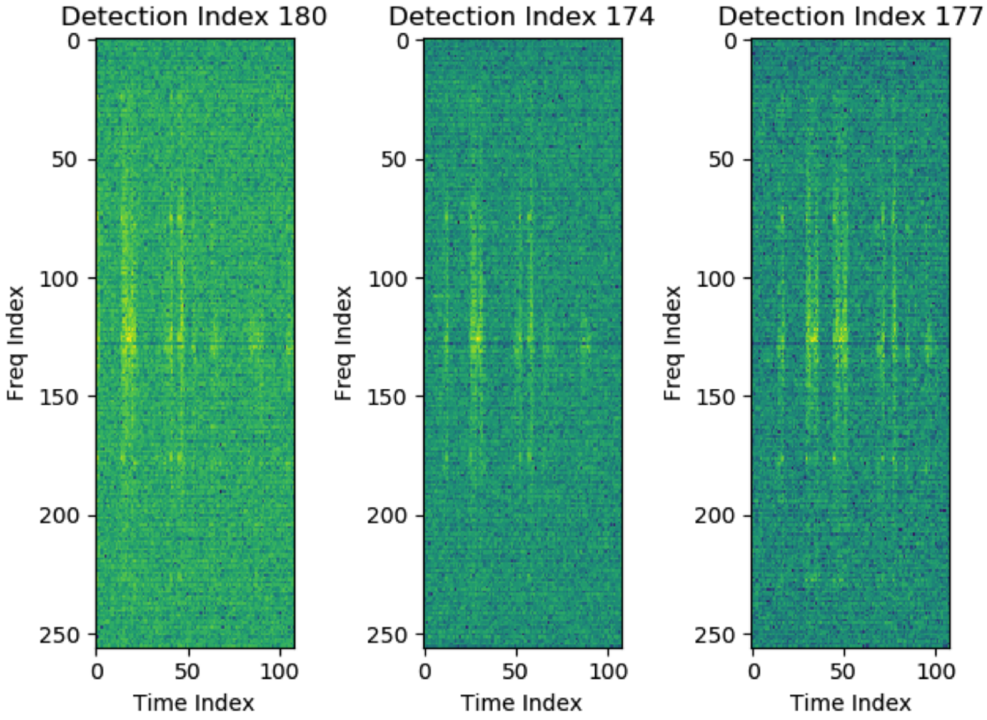


Figure 9.18: Cluster 5: $I = (1, 1)$**.**

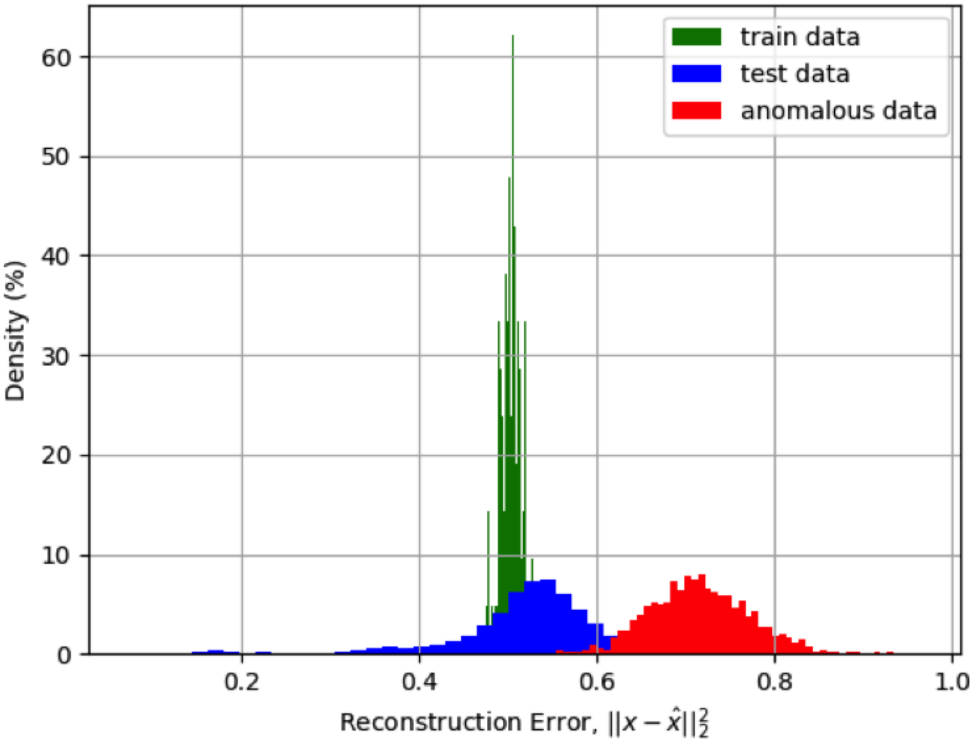


Figure 9.19: Auto-encoder training data.

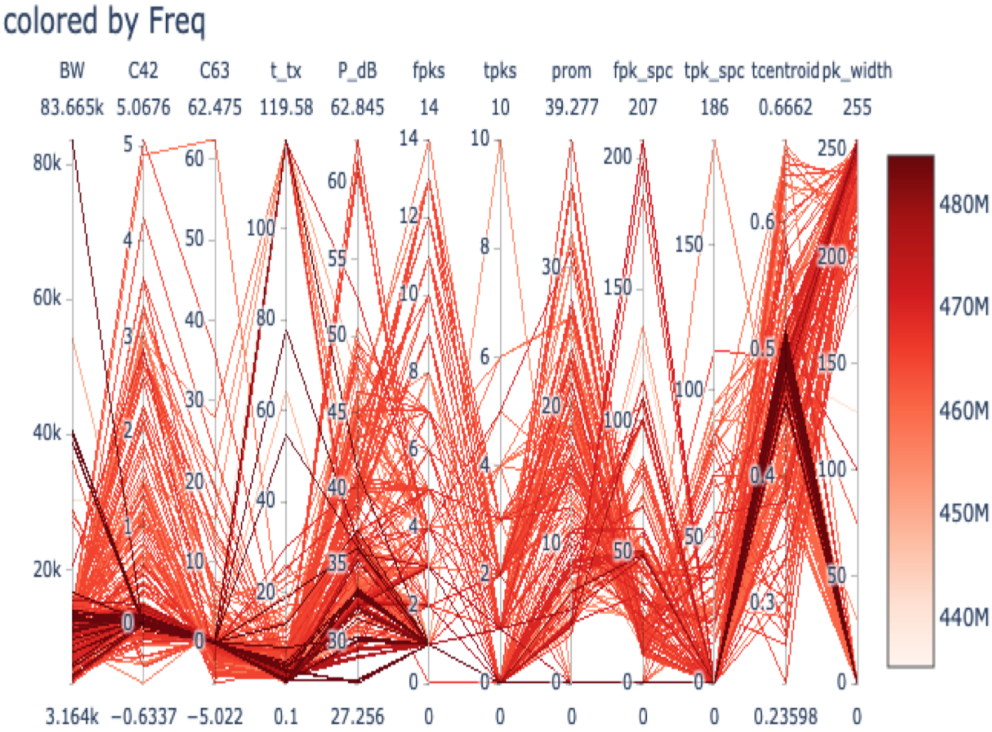


Figure 9.20: Auto-encoder full data.

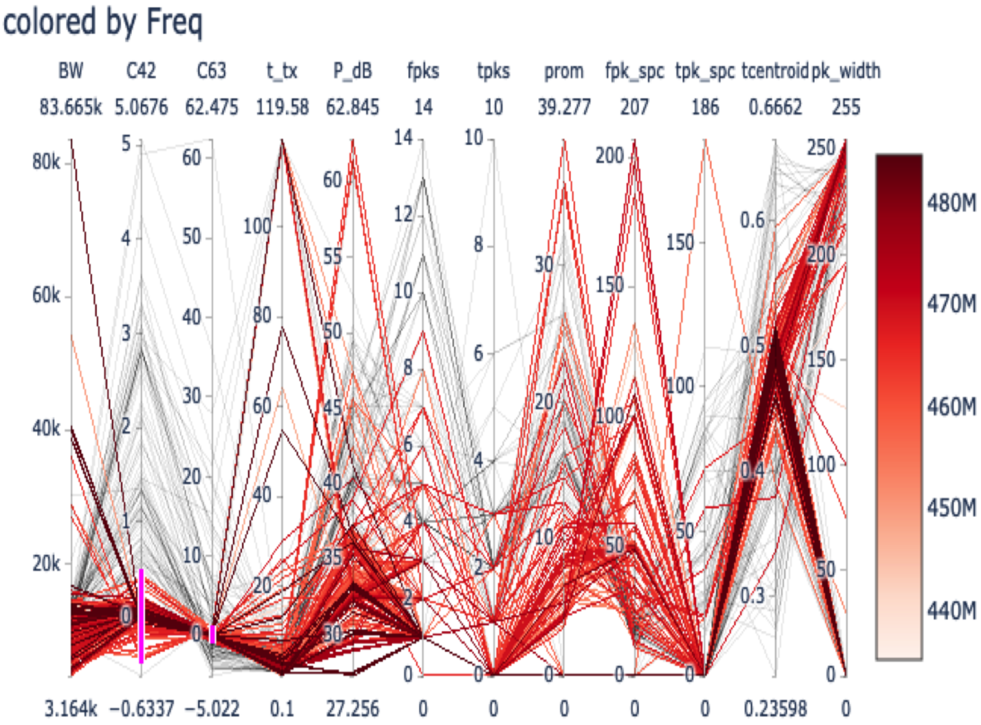


Figure 9.21: Auto-encoder training data.

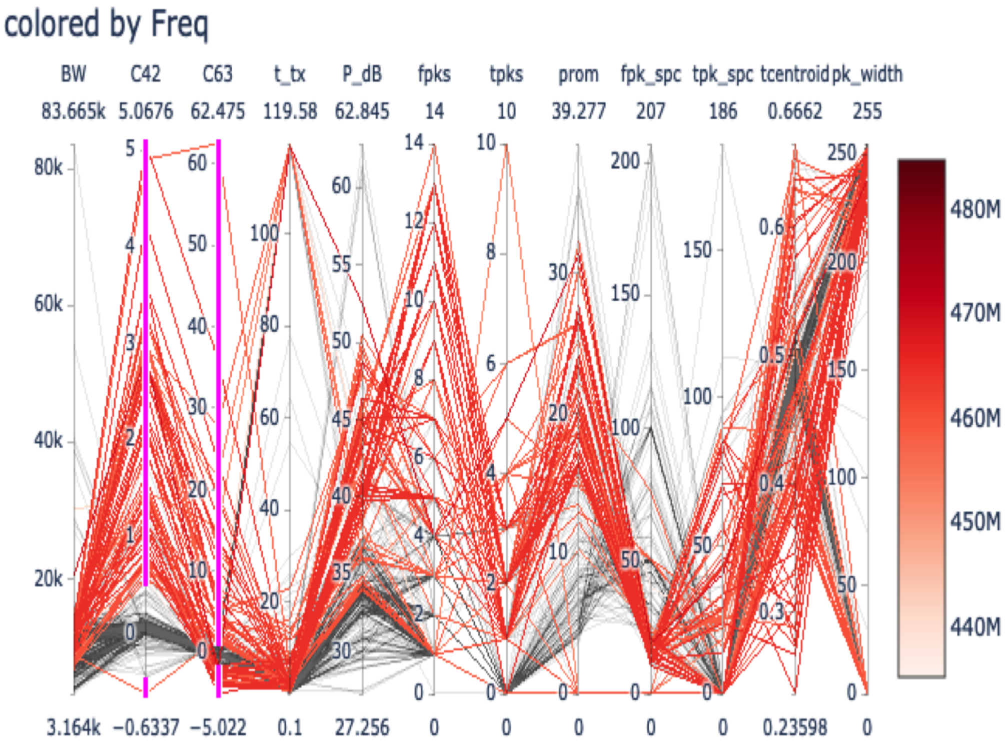


Figure 9.22: Auto-encoder testing data.

On the other hand, when the cumulants for each signal were combined with the extra information of the number of peaks in the time domain and the frequency domain, the resulting index I was sufficient to classify the incoming signals.

When tested on a subsequent dataset, this scheme detected a new cluster providing some confidence as to its future capability.

Future work naturally should include a full machine learning approach to classify the different signals. The index I above should then be used both to train the neural network (in terms of the clear classification label that it gives to each signal) and to benchmark the results of the machine learning algorithm.

Bibliography

[1] A. Abdelmutalab, K. Assaleh, M. El-Tarhmui, Automatic modulation classification based on high order cumulants and hierachical polynomial classifiers, Physical Communication, 21:10–18, 2016.

[2] https://www.mathworks.com/help/stats/dbscan.html