ivegotmail – compte-rendu – Classification Binaire de Spams

Ben Kirane Malik, Mouhoubi Fatima

2 mars 2018

Résumé

Nous nous intéressons au probléme de classification d'une base de courriers éléctroniques (emails). Nous souhaitons à partir du corps d'un email savoir s'il s'agit d'un spam ou non. Il s'agit d'inférer cette connaissance à partir d'un corpus d'emails avec une approche basée sur les probabilités (relation de Bayes). Cette approche est présentée en première partie. L'objet de ce compte-rendu est ensuite d'étudier différentes modélisations avec des descriptions enrichies pour répondre au problème de classification.

1 Classifieur par Inférence Bayésienne

Dans cette partie, pour établir à partir des distributions de probabilités estimées dans la phase d'apprentissage si un email est un spam ou non, nous illustrerons une solution á cette problématique avec un modèle très simple. Nous travaillerons tout le long sur deux phases distinctes : la phase d'apprentissage et la phase d'évaluation ou de prédiction pour discuter de la qualité du modèle choisis.

Il est naturel de s'intéresser à la description d'un email, description que nous noterons $D(x)$ ou $\hat{x}$, s'il s'agit d'un spam nous lui associerons une étiquette de valeur $+1$, sinon de valeur -1. Au départ avec un ensemble d'apprentissage $E = \{(x, y), y \in \{-1, +1\}\}$ et formellement, nous cherchons une application $f_{\Theta(E)}$ (classifieur), telle que pour tout email x non nécessairement dans E, $f_{\Theta(E)}(x)$ soit la classe de x.

La phase d'apprentissage est donc la phase où l'on estimera le modèle $\Theta(E)$ par inférence sur E et la phase d'évaluation est la phase oú on calcul l'image de $f_{\Theta(E)}$ pour un ensemble d'emails.

Nous considérons deux variables aléatoires :

1. X prendra les valeurs possibles des descriptions pour un email
2. Y vaut $+1$ ou bien -1 selon qu'il s'agit respectivement d'un spam ou non.

Notre approche est probabiliste : on souhaite par intuition connaître les distributions

$$P(Y = y|X = \hat{x}), \quad y \in \{-1, +1\} \tag{1}$$

et pouvoir les comparer pour espérer ensuite prédire si $D^{-1}(\hat{x})$ (l'email décrit par $\hat{x}$) est un spam ou encore si l'application $\hat{f}_E\colon Im(X) \to \{+1, -1\}$ apprise avec la description D indique pour un email x sa classe simplement par le calcul $\hat{f}_E[D(x)]$.

La frontière de décision peut se calculer en utilisant la relation de Bayes sur les probabilités conditionnelles (1)

$$\frac{P(X = \hat{x}|Y = +1)P(Y = +1)}{P(X = D(x))} - \frac{P(X = \hat{x}|Y = -1)P(Y = -1)}{P(X = D(x))} = 0 \tag{2}$$

(2) est l'estimation qu'un email décrit par $\hat{x}$ est un spam (resp. n'est pas un spam) pondérée par le ratio entre le nombre de spam (resp. non spam) et des emails de même description.

Nous considérons la description $D\colon x \mapsto l(x)$ tel que $l(x) \in \mathbb{N}$ est la longueur du corps d'un email. Nous estimons, pour cette description, étant donné une classe sur l'ensemble d'apprentissage, en comptant le nombre d'emails pour chaque description apparue lors du parcours de l'ensemble d'apprentissage. Il s'agit d'utiliser l'estimateur de fréquence pour la probabilité $P(X = D(x)|Y = y)$. Puis nous estimons la distribution de $P(Y = y)$ parce que l'ensemble d'apprentissage est fini. La distribution sur la classe des spams de notre estimateur est représentée sur la FIGURE 1. On lit la valeur pour $l(x)$ en abscisse et l'effectif en ordonnée.

FIGURE 1 – Estimation sur la longueur des spam

Plutôt que garder toutes les valeurs apparues pour $l(x)$ nous les regroupons dans des intervalles $[i_k, s_k]$ telles que pour tout intervalle k, $\text{Card}\left(\{\ x|\ l(x) \in [i_k, s_k\}\right) = 5$ par exemple (voir FIGURE 2).

FIGURE 2 – Estimation regroupée des distributions pour la description par longueur

Nous avons donc validé la phase d'apprentissage sur le corpus E et obtenu le modèle $\Theta(E)$ défini par deux fonctions : $e\colon \mathbb{N} \to [0,1]$ l'estimation de probabilité pour les email x décrits pour $l(x)$ sachant que x est un spam et $\overline{e}\colon \mathbb{N} \mapsto [0,1]$ l'estimateur sachant qu'il ne s'agit pas d'un spam ; et par $p = \text{Card}\left(\{(x,y) \in E \mid y = 1\}\right)/\text{Card}(E)$ (et $\overline{p} = 1 - p$).

Plus précisément, les fonctions e et $\overline{e}$ sont définies sur une partition finie de $\mathbb{N}$ comme indiqué pour construire les intervalles $[i_k, s_k]$ dont il est question plus haut. La prédiction s'évalue avec le signe de l'expression

$$e(\hat{x}) \cdot p - \overline{e}(\hat{x}) \cdot (1-p) \tag{3}$$

avec les notations vues jusque là. Donc en pratique on retournera l'estimation sur l'intervalle appris dans lequel $\hat{x} = l(x)$ se trouve. C'est justement ce qui a été implémenté pour ce cas de figure oú le descripteur D est une fonction à images dans $\mathbb{N}$.

Pour finir d'illustrer l'étude de ce premier classifieur, nous souhaitons valider ou infirmer ses prédictions. Pour ce faire nous divisons l'ensemble E d'apprentissage en deux sous-ensembles A et T. L'ensemble A pour la phase d'apprentissage et l'ensemble T pour estimer l'erreur des prédictions. Pour cette démonstration, nous choisissons arbitrairement 80% des emails étiquetés dans E pour constituer A et les emails restants constituent T. Ce choix arbitraire est alors réalisé 40 fois et à chaque découpage nous notons le nombre de prédictions incorrectes. La moyenne de la probabilité d'erreur et le score du meilleur classifieur sont présentés dans la Table 1.

Table 1 – Estimation de l'erreur

	$\min_f(P(f(x) \neq y))$	en moyenne
sur T	0.5454124189063948	0.5641682113067655
sur A	0.48148148148148145	0.515787037037037

Autant dire que les modèles appris avec une telle application de description ne sont pas fiables. Cependant, ce modèle nous a permis de détailler chaque phase dans la résolution du problème de classification posé. Nous allons nous porter plus longtemps à présent sur le choix de la description D pour compléter le manque d'informations qui nous empêche d'inférer correctement la classe à laquelle appartient un email.

2 Descriptions vectorielles d'emails, représentations, cas limites

L'objet d'étude, ici, est de réfléchir et d'évaluer la performance de modèles munis de descriptions enrichies.

Avec la description que nous avions choisi pour l'exemple de la première partie nous avons omis jusque là le corps de l'email. Il pourrait s'agir sur papier de décomposer le corps en mots et de représenter un email $x = \left(x_{i_1}, x_{i_2}, \ldots, x_{i_{l(x)}}\right)$ avec $\mathcal{D} = \{x_1, x_2, \ldots, x_d\}$ le dictionnaire des d mots possiblement dans un email. Nous pouvons imaginer plusieurs descriptions pour cette représentation de x :

$$\begin{aligned}
D_0(x) &= (\mathbb{I}_x[x_i])_{i \in \{1,\cdots,d\}} \\
D_1(x) &= (\mathbb{I}_x[x_i, x_j])_{i \neq j \in \{1,\cdots,d\}} \\
&\cdots \\
D_n(x) &= (\mathbb{I}_x[x_{i_1}, x_{i_2}, \ldots . x_{i_n}])_{i_1 \neq i_2 \neq \cdots \neq i_n \in \{1,\cdots,d\}}
\end{aligned}$$

où $\mathbb{I}_x[x_{i_1}, x_{i_2}, x_{i_n}]$ vaut 1 si x contient la sous-séquence de mots $(x_{i_1}, x_{i_2}, x_{i_n})$ et 0 sinon.

Par intuition, D_0 est une description moins riche que D_1, …, D_n, avec pour hypothèse très raisonnable que $n << d$. Mais il en est pas des moindres d'envisager de les implémenter. En effet la complexité pour représenter l'estimation $P(X = D_0(x)|Y)$ en mémoire est en $O(2^d)$ puisque nous représenterons les valeurs de probabilités pour les éventuelles 2^d descriptions dans $\{0,1\}^d$. Sur les corpus `spam`, `nospam` fournis, on dénombre 53808 mots. C'est à dire qu'il serait nécessaire d'avoir une structure de données donnant l'accès à 2^{53808} , et autrement dit environs $10^{1800 \cdot 9}$ flottants.

Face à ce problème nous pouvons ou bien étudier l'indépendance des variables aléatoires $X_i = \mathbb{I}_x[xi]$ et en conservant la description D_0 stocker l'estimation $P(X = D_0(x)|Y)$ avec $O(d)$ flottants $P(X_i = \mathbb{I}_x[x_i]|Y)$, ou bien réduire d à partir d'une heuristique parce que la borne 2^d reste très pessimiste.

Faute de quoi, on peut s'apercevoir qu'il n'est pas nécessaire de considérer $\mathcal{D}$ dans sa totalitée aussi bien qu'il n'est pas nécessaire de sauvegarder toutes les combinaisons possibles de parce que nombre d'entre elles sont nulles i.e. considérées comme impossibles au vu de la base d'apprentissage.

FIGURE 3 – Fréquences des mots dans le corpus

La FIGURE 2 propose une première heuristique qui est de ne pas considérer les mot trop peu apparus avec chacune des indicatrices au delà des 5000 premiers mots de $\mathcal{D}$ et de ne pas considérer les mots trop frèquents apparus dans l'ensemble d'apprentissage non plus. Dès à présent nous concentrerons nos efforts sur $D_{\sigma,\overline{\sigma}}$ la description pour un email sur le sous-ensemble significatif de $\mathcal{D}$, σ, pour lequel nous pourrions estimer non exhaustivement et raisonnablement les combinaisons apparues. Il s'agit de la restriction de D_0 avec σ et d'autre part prolongée sur le sous-ensemble non significatif $\overline{\sigma}$ sur lequel on présentera plus loin des alternatives sous des hypothèse faibles d'indépendance. On appel le couple $(\sigma, \overline{\sigma})$ la représentation du corpus.

Quand il s'agira de proposer une description dans ce contexte, nous distinguerons
— l'estimation basée sur le descripteur D_σ avec, par exemple σ les 500 mots les plus fréquents
— l'estimation basée sur les 500 premiers descripteurs $D_{i,\sigma}$ pour chacun de ces 500 mots de σ.

D'ailleurs nous nous proposons de visualiser empiriquement si le produit des distributions indépendantes des $P(X_i = D_{i,\sigma}(x)|Y)$ est proche de la distribution de $P(X = D_\sigma(x)|Y)$ pour un corpus donné. Nous réalisons autant d'estimations qu'il y a de blocs successifs de 500 mots dans $\mathcal{D}$ tronqué au mot 10000. Il s'agit de 20 blocs, la complexité empirique du nombre d'estimations pour chaque bloc se lit TABLE 2). Nous prenons les mots dans l'ordre des plus fréquents aux moins fréquents.

TABLE 2 – Évaluation empirique de la complexité de représentation

```
[0:10] 855, 782, 735, 674, 618, 563, 495, 481, 451, 408
[10:20] 380, 380, 337, 333, 335, 306, 272, 289, 259, 261
```

Le premier constat empirique est que la complexité de D_σ à été surestimée par 2^d dans ce cas pratique où $\mathrm{Card}(\sigma) = 500$. De plus, on visualise ici que l'information inférée sur le corpus est moindre sur les tranches de mots moins fréquents. Pour la représentation sur les 10000 premiers mots de $\mathcal{D}$, nous pourrions plutôt que considérer les estimations avec D_σ considérer les 20 estimations sur ces blocs avec D_{σ_i} où σ_i correspond à un bloc et supposer ces distributions indépendantes. Cette dernière hypothèse est moins forte que si on suppose l'indépendance sur des blocs de taille 1.

Le calcul des distributions pour chaque représentation σ_i, $1 \leq i \leq 20$, pour les 20 blocs de 500 mots et le calcul des distributions pour chaque représentation en supposant l'indépendance (donc 20 vecteurs de taille 500) sont réalisés séparéments en exploitant les features de la librairie `multiprocessing` de `python3` et en utilisant le module `numpy`. Nous sauvegardons ces fichiers qui, rappelons-le, décrivent l'ensemble des paramètres pour construire le modèle $\Theta(E)$.

En termes d'implémentation, l'estimation d'une représentation σ_i supposée indépendante est une liste de taille $\mathrm{Card}(\sigma_i)$ des probabilités $P(X = D_{j,\sigma_i}(x)|Y = y)$, $1 \leq j \leq \mathrm{Card}(\sigma_i)$, pour chaque label $y \in \{-1, +1\}$ et estimée sur les x de E. D'autre part, l'estimation d'une représentation σ_i est un dictionnaire dont les clés sont les chaînes de caractères de tailles $\mathrm{Card}(\sigma_i)$ composées de `0` et `1` qui pointent vers les probabilités $P(X = D_{\sigma_i}(x)|Y = y)$ – ces choix d'implémentation se justifient du fait que $D_{\sigma_i}(x) \in \{0,1\}^{\mathrm{Card}(\sigma_i)}$ et que $D_{j,\sigma_i}(x) \in \{0,1\}$.

Nous avons calculé pour chaque bloc la différence de probabilité sous chacune des hypothèses. Les différences sont tracées sur la FIGURE 4 et la FIGURE 5 si elle est inférieure à 10^{-2} pour les estimations des représentations σ_i, $1 \leq i \leq 19$. On en déduit que l'hypothèse d'indépendance des $X_j = D_{\sigma_i}$ est trop forte

FIGURE 4 – $\Delta_{+1}(x) = \left|\prod_j P(X = D_{j,\sigma_i}|Y = +1) - P(X = D_{\sigma_i}|Y = +1)\right|(x)$

FIGURE 5 – $\Delta_{-1}(x) = \left|\prod_j P(X = D_{j,\sigma_i}(x)|Y = -1) - P(X = D_{\sigma_i}(x)|Y = -1)\right|$

pour inférer un bon modèle surtout sur les blocs de mots les plus fréquents. On pourra se faire une idée plus fine des indépendances en fonction de σ_i le bloc de mots évalué avec la FIGURE 6 et la FIGURE 7 qui pour chaque classe et chaque bloc indiquent le nombre de clés pour lesquelles ont doit réfuter l'hypothèse pour un seuil de 10^{-2}.

FIGURE 6 – Seuils d'indépendances pour le corpus spam

Cette analyse nous conduit à ne surtout pas faire l'hypothèse d'indépendance sur les premiers descripteurs correspondants aux mots les plus fréquents.

À présent, nous avons posé les notations qui nous permettent de décrire les différentes représentations envisageables pour construire un modèle d'inférence sur un ensemble d'apprentissage et nous avons visité sur un corpus d'exemples d'apprentissages les écueils qui pourraient nuire à la qualité du modèle ou le rendre discutable pour sa complexité. Cependant, et nous l'avons observé plusieurs fois, restreindre la description à un sous-ensemble σ de mots n'est pas satisfaisant parce que l'on perds une quantité non négligeable d'information.

3 Descriptions étendues et représentations satisfaisantes

Figure 7 – Seuils d'indépendances pour le corpus nospam