

Data Mining

Disciplina: Machine Learning

Tema da Aula: Ensemble

Coordenação:

Prof. Dr. Adolpho Walter
Pimazzi Canton

Profa. Dra. Alessandra de
Ávila Montini

Prof. Carlos Eduardo Martins Relvas

Currículo

- Bacharel em Estatística, Universidade de São Paulo.
- Mestre em Estatística, Universidade de São Paulo.
- Itaú, 2010-2015. Principais atividades:
 - Consultoria estatística para várias áreas do banco com foco principal em melhorias no processo de modelagem de risco de crédito.
 - De 2013 a 2015, participação do projeto Big Data do banco usando tecnologia Hadoop e diversas técnicas de machine learning. Desenvolvemos diversos algoritmos em MapReduce usando R e Hadoop streaming, criando uma plataforma de modelagem estatística no Hadoop.
- Nubank, desde 2015. Principais atividades:
 - Equipe de Data Science, responsável por toda a parte de modelagem da empresa, desde modelos de crédito a identificar motivos de atendimento.

Conteúdo da Aula

- Ensemble
 - Averaging
 - Boosting
 - Voting
 - Stacking
- Text Mining

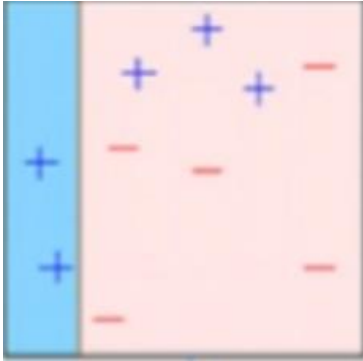
Ensembles

Ensembles

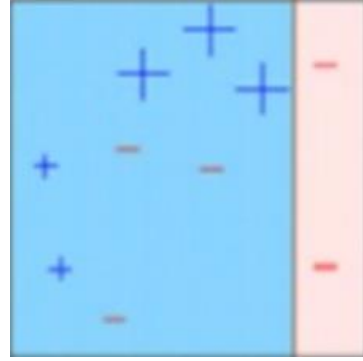
- Ensembles consiste de criar dois ou mais modelos preditivos correlacionados (mas não iguais) e combiná-los em um único modelo preditivo (melhorando a performance preditiva na maioria das vezes).
- Já vimos 2 exemplos de Ensembles (Random Forests e Boosting).

Ensembles

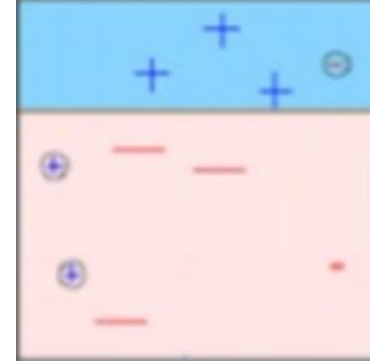
Modelo 1



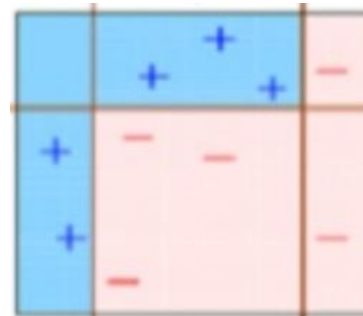
Modelo 2



Modelo 3



Modelo
Combinado



Ensembles

- Podemos combinar diferentes tipos de modelos, como por exemplo, uma Regressão Logística, um Random Forest, um Boosting e uma Rede Neural.



Ensembles

- "All models are wrong, but some are useful" – George Box
- "All models are wrong, some are useful, and their combination may be better" – Unknown.

How to win Kaggle

The big learning experience for me is how strong a team can be if the skills of its members complement each other. Rather like an ensemble in fact. None of us would have got in the top placings as individuals.

What we basically did was extract about 25-35 features from the original dataset, and applied an ensemble of five different methods; a regression random forest, a classification random forest, a feed-forward neural network with a single hidden layer, a gradient regression tree boosting algorithm, and a gradient classification tree boosting algorithm. The neural network was a pain to implement properly but improved things by a decent amount over the bagging and boosting based elements.

#17 | Posted 3 years ago



Alec Stephenson

Competition 1st | Overall 642nd

Posts 82 | Votes 55

Joined 1 Sep '10 | Email User

[Permalink](#)

How to win Kaggle

I used an ensemble of 15 models including GBMs, weighted GBMs, Random Forest, balanced Random Forest, GAM, weighted GAM (all with bernoulli/binomial error), SVM and bagged ensemble of SVMs.

I haven't try to fine tune each models individually but looked for diversity of fits.

My best score (0.89345, not in the private leaderboard as I haven't selected it in my final set) was an ensemble of 11 models which excluded the SVMs fits.

#18 | Posted 3 years ago



Xavier Conort

Competition 2nd | Overall 33rd

Posts 49 | Votes 94

Joined 23 Sep '11 | Email User

[Permalink](#)

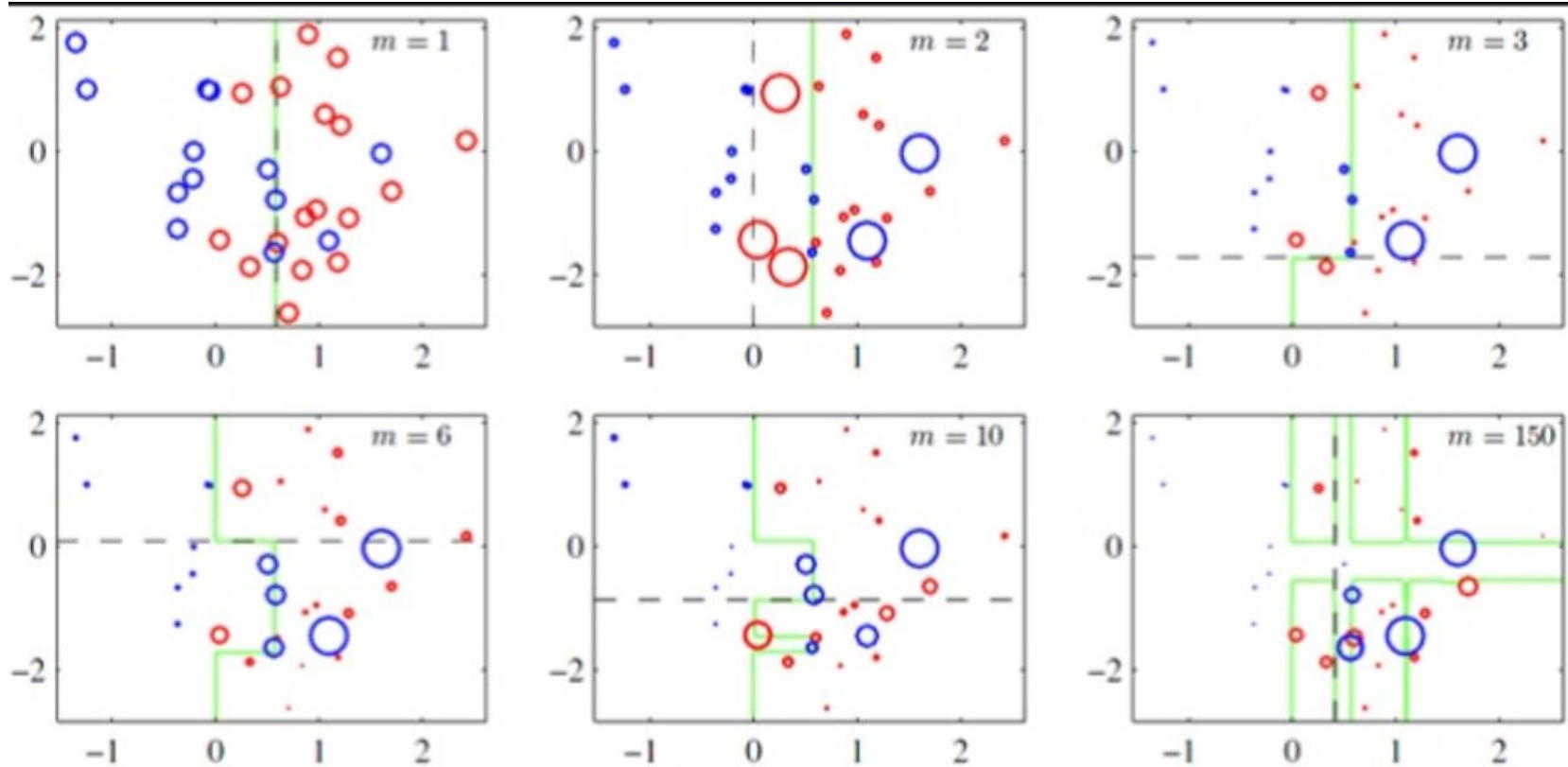
Averaging

- Treine vários modelos independentes e retorne a médias das previsões.
- Reduz variância e com isso tende a aumentar a acurácia.
- Robusto contra outliers.
- Geralmente usado com árvores (Random Forest).

Boosting

- Também reduz variância e com isso tende a aumentar a acurácia.
- Não robusto a outliers (usa todos os dados).
- Flexível, pode ser usado com qualquer função de perda.
- Geralmente usado com árvores (Boosting Trees).

Boosting



Voting

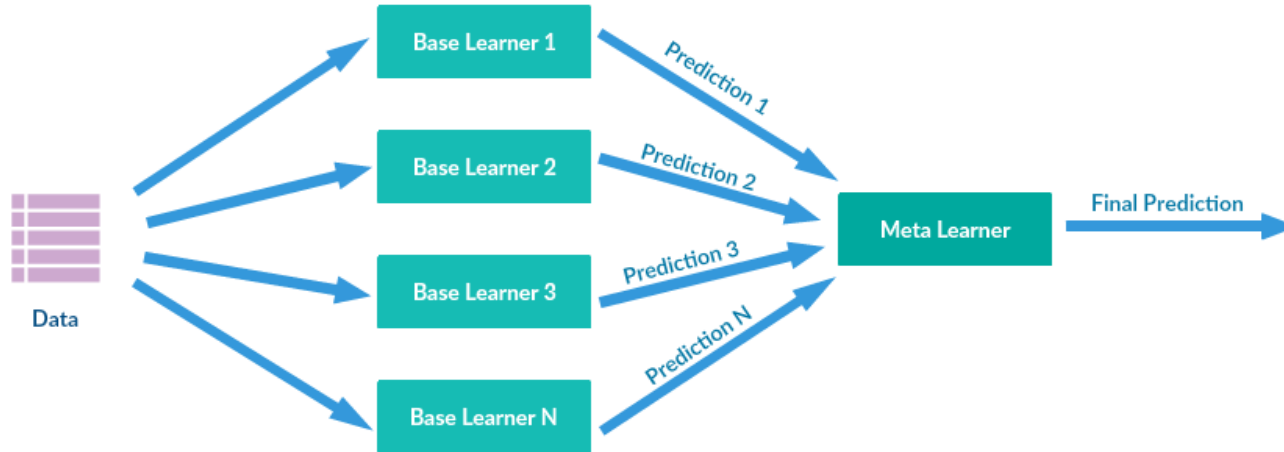
- Construa diferentes modelos e então use como resultado o voto majoritário ou o chamado o soft vote (probabilidade média).
- Funciona como uma democracia. Prós e cons?
- Não é robusto a outliers.

Stacking

- Uma das técnicas de combinação de modelos mais utilizadas hoje, sendo a preferida do Kagglers.
- Consiste em criar diversos modelos chamados de base learners e depois criar um novo modelo usando as previsões prévias como input.

Stacking

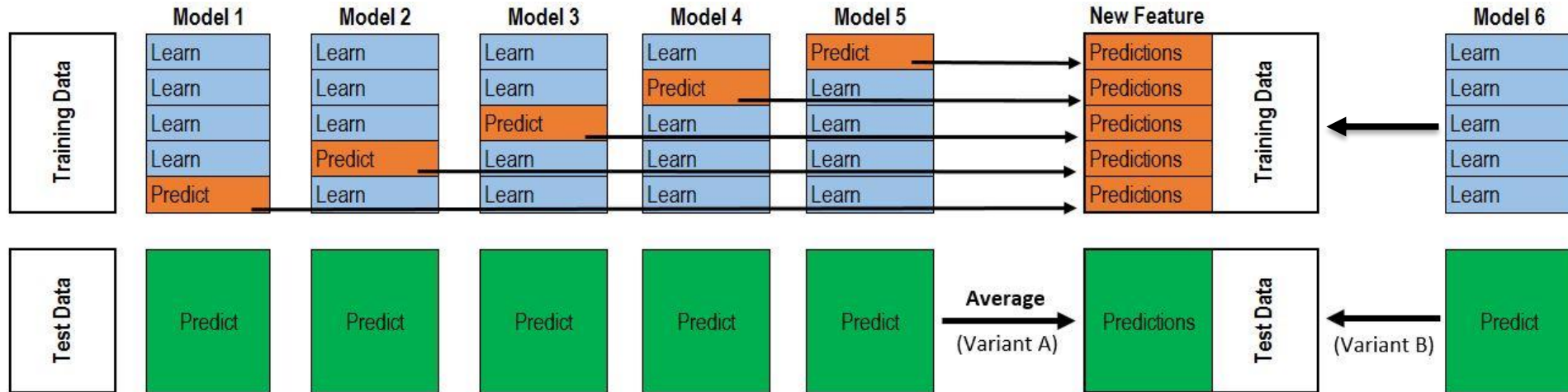
- Exemplo de stacking com um nível.



Stacking

- Geralmente utiliza-se apenas um nível, mas pode estender o raciocínio para vários níveis.
- Como evitar que o overfitting no nível anterior se espalhe?
- **Não podemos utilizar os mesmos dados de treino.**
- Assim, podemos fazer o novo modelo na base de validação ou utilizando uma estratégia de cross-validation.

Stacking



Ensembles - Laboratório

Base de Iris

- Base com 150 observações e 5 variáveis.

Comprimento da sépala

Largura da sépala

Comprimento da pétala

Largura da pétala

Espécie



- Disponível em: <https://archive.ics.uci.edu/ml/datasets/iris>

Ensembles - Exercício

Laboratório R - Base de Spam

- Base com 4.601 e-mails. Porcentual em que 54 palavras ou pontuações aparecem em cada e-mail. Além disso, temos o tamanho médio das palavras, tamanho da maior palavra e quantidade de palavras.

Objetivo:

Criar um detector automático de SPAM que verificará cada novo e-mail.

- Disponível em: <https://archive.ics.uci.edu/ml/datasets/Spambase>

Natural Language Processing

Campo de Estudo da computação cujo objetivo é pesquisar sobre formas de interação entre os computadores e escrita humana (*human-computer interaction*).

Os maiores desafios de NLP (natural language processing) envolvem: entender textos, fazer com que computadores derivem significados de textos e até mesmo desenvolver textos.

Os algoritmos mais antigos eram baseados em conceitos léxicos. Já os algoritms modernos de NLP são baseados em machine learning.

Análise de Sentimentos

- Análise de Sentimentos ou Opinion Mining é o estudo computacional de opiniões, sentimentos e emoções expressos em texto.

Bing Liu, 2010.

- Conjunto de técnicas que nos permite mapear atitudes e sentimentos na internet por meio de posts em blog, comentários, reviews e tweets de diferentes tópicos.
- Podendo assim verificar a preferência dos usuários a produtos, marcas, propagandas específicas, etc.

Análise de Sentimentos – Métodos Léxicos

- Dicionário:
definir lista de palavras positivas e negativas (incluir sinônimos e antônimos). Assim, atribuir um escore a cada texto.
- Corpus based:
similar a técnica do dicionário, mas automaticamente expande adjetivos combinados.
Ex.: “este carro é grande e espaçoso.” Se grande estava como positivo, marcamos espaçoso como positivo também.

Análise de Sentimentos

Objetivos:

- Construir algoritmo que “leia” novas críticas / comentários / twitters e classifique como positivo / negativo.
- Criar métricas para monitoramento de marca / produto em tempo real com a mínima intervenção humana.
- De forma praticamente instantânea responder a crises. (Reputation Management);

Análise de Sentimentos




Positivos

Análise de Sentimentos

Negativos


060 out of 1930 people found the following review useful:

 **A much greater improvement on MOS 5X better.**
 ★★★★★★
 Author: [corde1](#) from Washington State, USA
 19 March 2016

*** This review may contain spoilers ***


I am drawn back to when this movie was first announced and how I despised the notion of a Batman V Superman film a la Freddy vs Jason, Alien vs Predator and so on. It all seemed like a huge gimmick to me and I expected the worst for this movie. Fast forward a couple years, I am sitting down at a private screening, the lights go down and the movie opens up. The opening sequence was not what I expected but shows Snyder's visual creativity in a similar manner as he did with the opening sequence of Watchmen.

13 out of 1099 people found the following review useful:

 **Ben Affleck!!!**
 ★★★★★★
 Author: [fero_king-65322](#) from United States
 19 March 2016


*** This review may contain spoilers ***

Affleck as Bruce Wayne/Batman was pure awesome and dark. You can see he's traumatized by the events in the past, and he has some fantastic lines in general (Yes we see the whole parents thing again). Nobody should really be surprised that he was good with the role considering he's a solid actor, but I thought he was great. Versus Bale? That's an interesting question, because Affleck definitely regards Batman as a permanent part of his persona, while in Nolan series he eventually doesn't want Batman. In combat he's pretty much everything everyone would hope for from a film with Batman. One comment on the latest BvS trailer said he fought like the Arkham video games from the scene they showed. That's a good comparison. He's incredible in fights. Also the way Snyder records him in action, you can actually see him fighting unlike the Nolan films, which I'd imagine would make many people happy. He also gets a chance to use his detective skills, do not miss that. Also his relationship with Batman is great. Alfred is just needed in general, because more attention to the role (the Batman from for

 **A complete package**
 ★★★★★★
 Author: [gadme](#) from United Kingdom
 20 March 2016


*** This review may contain spoilers ***

Boy oh Boy....this was a great experience....I have to say after Man of Steel (which was okay but by no means great), I didn't have much expectations from this, and to top it off the castings for this film had set the comic book fans on fire. But tell you what, this is epic. Some people are even calling it better than the Dark Knight, while I wouldn't go that far, I would still call this a complete package. Affleck's Batman AMAZING. His Bruce Wayne is not as cool and casual as Christian Bale's but he doesn't need to considering this is a relatively old Batman who has seen some very dark times. His partnership with Jeremy Irons adds some much needed lightness to the movie. Cavill delivered the dialogues during the fight scene with Batman in a near perfect way. Gal Gadot had a much bigger role to play than seen in the trailers. And

 **Argh.**
 ★★★★★★
 Author: [Typpo](#) from United States
 16 April 2016


*** This review may contain spoilers ***

This was a very frustrating movie. I had been looking forward to it, although was initially skeptical after hearing of Ben Seeing the critics views at RottenTomatoes was very discouraging, but for some reason, IMDb showed favorable reviews gave me some hope.

 **Do not pay for this!**
 ★★★★★★
 Author: [totakos](#) from Hungary
 24 March 2016

*** This review may contain spoilers ***

The storyline was slow, and the whole movie was built for the 10 years old children to understand what's happening. It was filled with clichés and was not creative at all. I was so disappointed after watching this. I watched this with my friends and they are on the same opinion. The worst Batman / Superman movie ever I think. Ben Affleck was not a good choice for Batman because he is not "blue blooded" enough and was not able to act like them.

 **Yawn of Justice, Yet another Hollywood Hack job**
 ★★★★★★
 Author: [dashoost](#) ([dashoost@gmail.com](#)) from Earth
 26 March 2016

*** This review may contain spoilers ***

SPOILER ALERT !!! (Funny how there can still be such a thing when the movie is already spoiled to begin with)

This movie was marketed to patrons of all ages but only appeals to ages 12 or younger.

My top 8 reasons (in no discernible order) to avoid seeing this movie and save 2 1/2 hours of your life...Don't do it!

1. Amy Adams/ Lois Lane - Wow! what a terribly written part for this character...Clueless and painfully unaffected by everything going on around her... I speak for all true Superman fans when I say, Margot Kidder, eat your heart out baby!
2. The opening sequence - Bat boy Bruce Wayne sees his parents killed at gunpoint....AGAIN! Why must we always have to be spoon-fed the origin of this character in every damn movie. The psychotic flashbacks from his past will likely be thrown in later on in the movie and can allude to the dark knights brooding demeanor just fine, thank you.

Análise de Sentimentos

Desafios: A linguagem é ambígua! O contexto é importante.

"I bought an iPhone a few days ago. It is such a nice phone. The touch screen is really cool. The voice quality is clear too. It is much better than my old Blackberry, which was a terrible phone and so difficult to type with its tiny keys. However, my mother was mad with me as I did not tell her before I bought the phone. She also thought the phone was too expensive, ..."

Análise de Sentimentos

Desafios: Ironia

- Muito obrigado banco XXX por me cobrar esta multa e este juros.
- 2 horas sem sinal no celular. Excelente serviço YYY.

Análise de Sentimentos

Desafios: Diferentes tipos de negação

- Direta: “Não gostei deste telefone.”
- Ambígua: “Não só não gostei, como também achei caro.”
- Indireta: “Talvez seja muito bom, só não percebi.”

Análise de Sentimentos

Desafios: Gírias (Mudança no tempo) e Erros de gramática.

- Como entender a palavra “bomito”?

Desafios: Co-referências.

- Ontem assinei o Netflix com meu Nubank. Ótima experiência!

O que? O Nubank ou o Netflix? Ou os dois?

Análise de Sentimentos

Desafios: Contexto!!

- A bateria deste celular demora para acabar. (Positivo)
- A fila de espera do restaurante demora para chamar. (Negativo)

Bag of Words

A maioria dos algoritmos mais modernos de análise de sentimento utiliza-se da técnica bag of words. Esta técnica consiste em, a partir de um texto, criar variáveis explicativas a partir da frequência de cada palavra.

Ex: 3 comentários:

- | | |
|---|-----------|
| 1) O Masp é muito bom. | Positivo. |
| 2) O atendimento do Masp tem poucos funcionários. | Negativo. |
| 3) O Masp não tem um preço bom. | Negativo. |

Bag of Words

→ Lista de palavras:

[muito, bom, atendimento, poucos, funcionários, não, preço].

	X	Y
1)	[1,1,0,0,0,0,0]	1
2)	[0,0,1,1,1,0,0]	0
3)	[0,1,0,0,0,1,1]	0

Pré-processamento – Stopwords

É recomendado remover as chamadas stopwords como:

- That, the, and, ...
- Ou, e, o, a, de, ...

Os principais pacotes já possuem uma lista de stopwords para ser removido (em português também!).

Pré-processamento – Stemming

Stemming é o processo de reduzir cada palavra para sua raíz.

- Dormir, dormindo, dorme, dormiu → dorme
- Cats, catlike, and catty → cat

Pré-processamento – Tokenization

Stemming é o processo de quebrar um texto em palavras, frases, símbolos ou outros elementos com significado.

Term Frequency – Inverse Document Frequency (TF-IDF)

TF-IDF é uma estatística usada para refletir o quão importante um token é em relação a um documento.

É frequentemente usada como um peso em text mining.

$$\text{TF weight (of the token in the document)} = \frac{\text{Number of times the token appears in the document}}{\text{Total Number of Tokens in the document}}$$

$$IDF(t) = \frac{\text{Total number of documents}}{\text{Number of documents in U that contain t}} = \frac{N}{n(t)}$$

$$TF\text{-}IDF = TF \times IDF$$

Naive Bayes

Teorema da Bayes:

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{P(B|A)P(A)}{P(B)}$$

Naive Bayes para análise de sentimento:

X representa o vetor de frequência de palavras (originário do Bag of Words)

Y representa a classe (positivo, negativo, neutro, etc...).

Naive Bayes

\mathbf{X} representa o vetor de frequência de palavras (originário do Bag of Words) de tamanho n

Y representa a classe (positivo, negativo, neutro, etc...).

$$\begin{aligned} P(Y = y | \mathbf{X}) &= \frac{P(\mathbf{X} | Y = y) P(Y = y)}{P(\mathbf{X})} \\ &= \frac{P(X_1 | Y = y) \dots P(X_n | Y = y) P(Y = y)}{P(\mathbf{X})} \\ &\approx P(X_1 | Y = y) \dots P(X_n | Y = y) P(Y = y) \end{aligned}$$

Naive Bayes

$$P(Y = \mathbf{y}|X) \approx P(X_1|Y = \mathbf{y}) \dots P(X_n|Y = \mathbf{y})P(Y = \mathbf{y})$$

- $P(Y = \mathbf{y})$, probabilidade à priori de cada class (proporção observada).
- $P(X_1|Y = \mathbf{y}) = \frac{\#X_{1,\mathbf{y}}}{\sum_w \#w,\mathbf{y}}$, fração de vezes que a palavra X_1 aparece na classe \mathbf{y} sobre a soma de todas as outras palavras da mesma classe.

Naive Bayes

$$P(Y = \mathbf{y}|X) \approx P(X_1|Y = \mathbf{y}) \dots P(X_n|Y = \mathbf{y})P(Y = \mathbf{y})$$

- $P(Y = \mathbf{y})$, probabilidade à priori de cada class (proporção observada).
- $P(X_1|Y = \mathbf{y}) = \frac{\#X_{1,\mathbf{y}}}{\sum_w \#w,\mathbf{y}}$, fração de vezes que a palavra X_1 aparece na classe \mathbf{y} sobre a soma de todas as outras palavras da mesma classe.

Problema quando nunca observamos uma palavra em uma determinada classe, o que resultaria em probabilidade 0.

Naive Bayes

Assim, usamos as seguinte fórmulas:

- $$P(X_1|Y = y) = \frac{\#X_{1,y+1}}{\sum_w(\#w,y+1)}$$

Naive Bayes

Exemplo:

Treino:	1.) Gostei	Positivo	1	$X_1=[1,0,0]$
	2.) Quase odiei	Negativo	0	$X_2=[0,1,1]$
	3.) Gostei	Positivo	1	$X_3=[1,0,0]$
Teste:	4.) Quase gostei			$X_4=[1,1,0]$

Naive Bayes

Teste: 4.) Quase gostei

$$X_4=[1,1,0]$$

$$P(Y = 1) = \frac{2}{3}, P(Y = 0) = \frac{1}{3}$$

$$P(\text{Gostei}|Y = 1) = \frac{2+1}{2+3} = \frac{3}{5}, P(\text{Gostei}|Y = 0) = \frac{0+1}{2+3} = \frac{1}{5}$$

$$P(\text{Quase}|Y = 1) = \frac{0+1}{2+3} = \frac{1}{5}, P(\text{Quase}|Y = 0) = \frac{1+1}{2+3} = \frac{2}{5}$$

$$P(Y = 1|X_4) = \frac{1}{5} \frac{3}{5} \frac{2}{3} = 0.08$$

$$P(Y = 0|X_4) = \frac{1}{5} \frac{2}{5} \frac{1}{3} = 0.026$$

Naive Bayes

Pros:

- Fácil e rápido para treinar e predizer.
- Indicado na presença de variáveis categóricas.
- Não é sensível a variáveis irrelevantes.

Cons:

- Suposição de independência é muito forte e difícil de se encontrar na prática.

Ferramentas Úteis

AFINN: lista de palavras em inglês classificadas de -5 (negativas) a 5 (positivas).

WordNet: banco de dados léxico que agrupa conjunto de sinônimos para cada palavra inglesa (distingue entre verbos, substantivos, adjetivos e advérbios).

SentiWordNet: extensão do WordNet que adiciona uma classificação para cada conjunto de sinônimos.

Aplicação

Base de Dados

Criamos a partir do Twitter. Colhemos 3200 twitters em inglês com a palavra **Samsung** entre os dias 04/10/2016 e 11/10/2016.

```
1 install.packages('twitter', dependencies=TRUE, repos='http://cran.rstudio.com/')
2 install.packages('plyr', dependencies=TRUE, repos='http://cran.rstudio.com/')
3 install.packages('stringr', dependencies=TRUE, repos='http://cran.rstudio.com/')
4 install.packages('ggplot2', dependencies=TRUE, repos='http://cran.rstudio.com/')
5 install.packages('RTextTools', dependencies=TRUE, repos='http://cran.rstudio.com/')
6 install.packages('e1071', dependencies=TRUE, repos='http://cran.rstudio.com/')
7
8 library(twitter)
9 library(plyr)
10 library(stringr)
11 library(ggplot2)
12 library(e1071)
13 library(RTextTools)
```

Base de Dados

Criando a conexão do R com o Twitter.

```
21 consumerKey <- "user--consumer--key"
22 consumerSecret <- "user--consumer--Secret"
23 accessToken <- "access--token--key"
24 accessTokenSecret <- "access--token--Secret"
25
26 setup_twitter_oauth(consumerKey, consumerSecret,
27                     access_token = accessToken,
28                     access_secret = accessTokenSecret)
```

Onde encontrar as chaves de conexão: <https://apps.twitter.com/>

Base de Dados

400 tweets de cada dia. Poderíamos pedir 3200 tweets mudando os parâmetros since a until, mas não garantiríamos uma distribuição uniforme entre os dias.

```
25 list1 <- searchTwitter('Samsung',n=400, lang="en",
26                        since='2016-10-04', until ='2016-10-05')
27 list2 <- searchTwitter('Samsung',n=400, lang="en",
28                        since='2016-10-05', until ='2016-10-06')
29 list3 <- searchTwitter('Samsung',n=400, lang="en",
30                        since='2016-10-06', until ='2016-10-07')
31 list4 <- searchTwitter('Samsung',n=400, lang="en",
32                        since='2016-10-07', until ='2016-10-08')
33 list5 <- searchTwitter('Samsung',n=400, lang="en",
34                        since='2016-10-08', until ='2016-10-09')
35 list6 <- searchTwitter('Samsung',n=400, lang="en",
36                        since='2016-10-09', until ='2016-10-10')
37 list7 <- searchTwitter('Samsung',n=400, lang="en",
38                        since='2016-10-10', until ='2016-10-11')
39 list8 <- searchTwitter('Samsung',n=400, lang="en")
40
41 list <- c(list1, list2, list3, list4, list5, list6, list7, list8)
```

Base de Dados

- Transformando em dataframe.
- Convertendo o formato da data.
- Salvando o dataframe em um arquivo CSV.

```
43 df <- twListToDF(list)
44 df <- df[, order(names(df))]
45 df$created <- strptime(df$created, '%Y-%m-%d')
46
47 write.csv(df, "/Users/carlos/Desktop/twitter_samsung.csv", row.names = F)
```

Base de Dados

Função para classificar o tweet como positivo ou negativo. Qual o problema com isso? Resumo da função:

- Recebe como argumentos os tweets, a lista de palavras positivas e a lista de palavras negativas.
- Para cada tweet, removemos caracteres especiais, pontuações, números, etc (todos os elementos não alfa-númericos).
- Transformamos para minúsculo e separamos as palavras por espaço.
- Checamos quantas palavras positivas e negativas cada tweet tem. Assim, criamos um score para definir o que é positivo ou negativo.

Base de Dados

- Função para classificar o tweet como positivo ou negativo.

```
51 > score.sentiment <- function(sentences, pos.words, neg.words){  
52 >   scores <- lapply(sentences, function(sentence, pos.words, neg.words){  
53     sentence <- gsub('[:punct:]', '', sentence)  
54     sentence <- gsub('[:cntrl:]', '', sentence)  
55     sentence <- gsub('\\d+', '', sentence)  
56     sentence <- str_replace_all(sentence, "[^[:alnum:]]", " ")  
57     sentence <- tolower(sentence)  
58     word.list <- str_split(sentence, '\\s+')  
59     words <- unlist(word.list)  
60     pos.matches <- match(words, pos.words)  
61     neg.matches <- match(words, neg.words)  
62     pos.matches <- !is.na(pos.matches)  
63     neg.matches <- !is.na(neg.matches)  
64     score <- sum(pos.matches) - sum(neg.matches)  
65     return(score)  
66   }, pos.words, neg.words)|  
67   scores.df <- data.frame(score=scores, text=sentences)  
68   return(scores.df)  
69 }
```

Base de Dados

- Lendo as listas de palavras positivas e negativas.
- Aplicando a função para a criação do score de “positividade”.

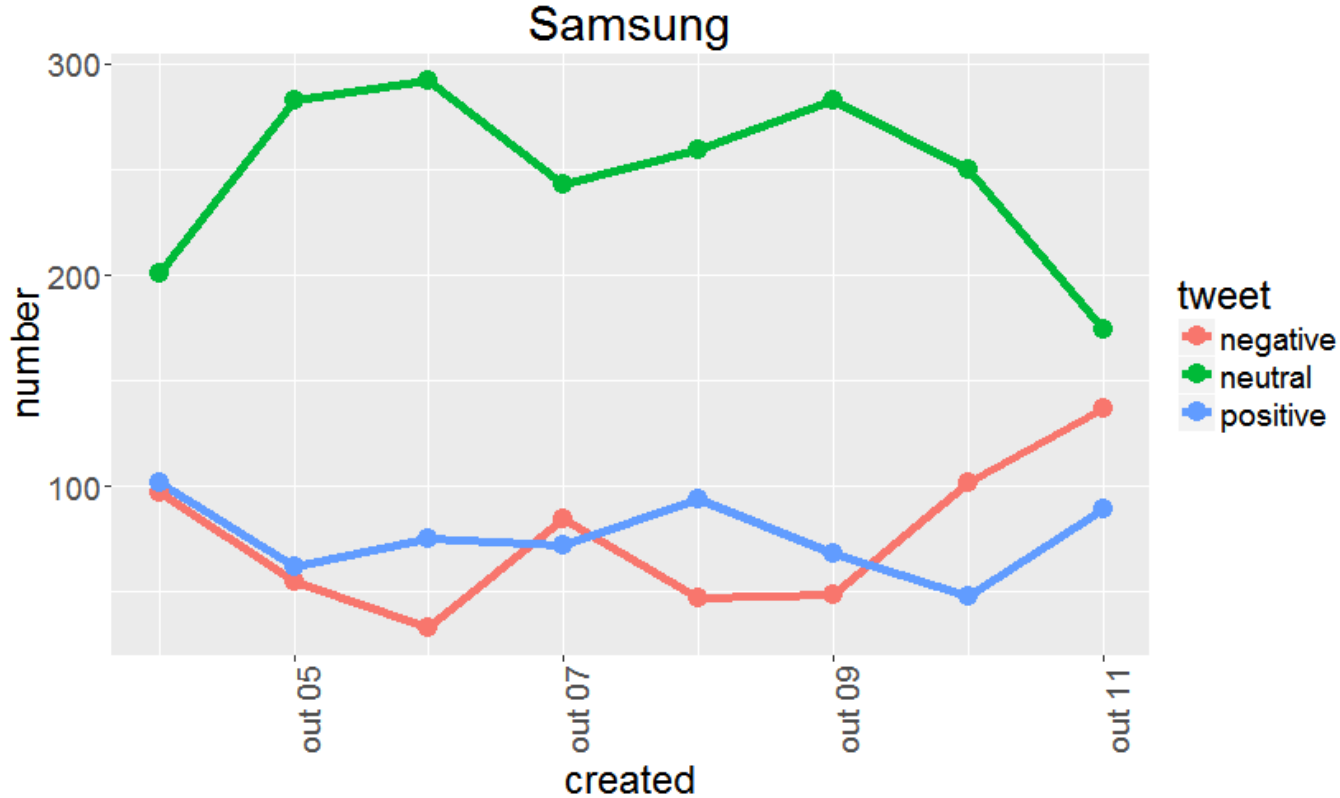
```
71 pos <- scan('positive_words.txt', what='character', comment.char=';')
72 neg <- scan('negative_words.txt', what='character', comment.char=';')
73 pos.words <- c(pos, 'upgrade')
74 neg.words <- c(neg, 'wtf', 'wait', 'waiting', 'epicfail')
75 Dataset <- df
76 Dataset$text <- as.factor(Dataset$text)
77 scores <- score.sentiment(Dataset$text, pos.words, neg.words)
```

Base de Dados

- Categorizando os tweets como positivo (score > 0), negativo (score < 0) ou neutro (score = 0).
- Sumarizando por data a contagem de tweets bons / maus.
- Criando o gráfico com o ggplot.

```
79 stat <- scores
80 stat$created <- df$created
81 stat$created <- as.Date(stat$created)
82 stat <- mutate(stat, tweet=ifelse(stat$score > 0, 'positive',
83                                   ifelse(stat$score < 0, 'negative', 'neutral'))))
84 by.tweet <- group_by(stat, tweet, created)
85 by.tweet <- summarise(by.tweet, number=n())
86
87 ggplot(by.tweet, aes(created, number)) + geom_line(aes(group=tweet, color=tweet), size=2) +
88   geom_point(aes(group=tweet, color=tweet), size=4) +
89   theme(text = element_text(size=18), axis.text.x = element_text(angle=90, vjust=1)) +
90   ggtitle('Samsung')
```


Base de Dados



- Aumento de tweets negativos com as notícias do Galaxy Note 7

Pré-modelagem

- Remover os tweets neutros.
- Retirar os caracteres especiais do tweet, do mesmo modo que fizemos anteriormente.

```
95 id_notneutral <- stat[,4] %in% c('positive', 'negative')
96 data_model <- stat[id_notneutral,]
97
98 sentence <- data_model[,2]
99 sentence <- gsub('[[punct:]]', '', sentence)
100 sentence <- gsub('[[cntrl:]]', '', sentence)
101 sentence <- gsub('\\d+', '', sentence)
102 sentence <- str_replace_all(sentence, "[^[:alnum:]]", " ")
103 sentence <- tolower(sentence)
```

Bag of Words

- Fixar a semente e quebrar os tweets entre treino (80%) e teste (20%).
- Criar a matrix de Bag-of-words, removendo “Stop words”.
- Vamos ver o help da função.

```
105 set.seed(42)
106 id_train <- sample(1:nrow(data_model), 0.8*nrow(data_model), replace = F)
107
108 mat= create_matrix(sentence, language="english",
109                   removeStopwords=TRUE, removeNumbers=TRUE,
110                   stemwords=FALSE)
111
112 mat = as.matrix(mat)
```

Naive Bayes

- Vamos executar o Naive Bayes com a matriz de Bag of Words criada.
- Checar a performance no treino e teste.

```
116 classifier = naiveBayes(mat[id_train,], as.factor(data_model[id_train,4]))
117 predicted = predict(classifier, mat)
118
119 table(data_model[id_train,4], predicted[id_train])
120 table(data_model[-id_train,4], predicted[-id_train])
```

```
> table(data_model[id_train,4], predicted[id_train])
```

	negative	positive
negative	263	223
positive	452	34

```
> table(data_model[-id_train,4], predicted[-id_train])
```

	negative	positive
negative	95	24
positive	113	11

Árvore de Decisão

- Testar árvores de decisão com diferentes profundidades.
- Escolher a melhor árvore de acordo com a base de teste.

```
126 dados <- data.frame(cbind(data_model[,4], as.matrix(mat)))
127 names(dados)[1] <- c('Sentimento')
128
129 fit3 <- ctree(as.factor(Sentimento) ~ ., data=dados[id_train,],
130               controls = ctree_control(maxdepth = 3))
131 fit5 <- ctree(as.factor(Sentimento) ~ ., data=dados[id_train,],
132               controls = ctree_control(maxdepth = 5))
133 fit10 <- ctree(as.factor(Sentimento) ~ ., data=dados[id_train,],
134                controls = ctree_control(maxdepth = 10))
135 fit <- ctree(as.factor(Sentimento) ~ ., data=dados[id_train,])
136
```

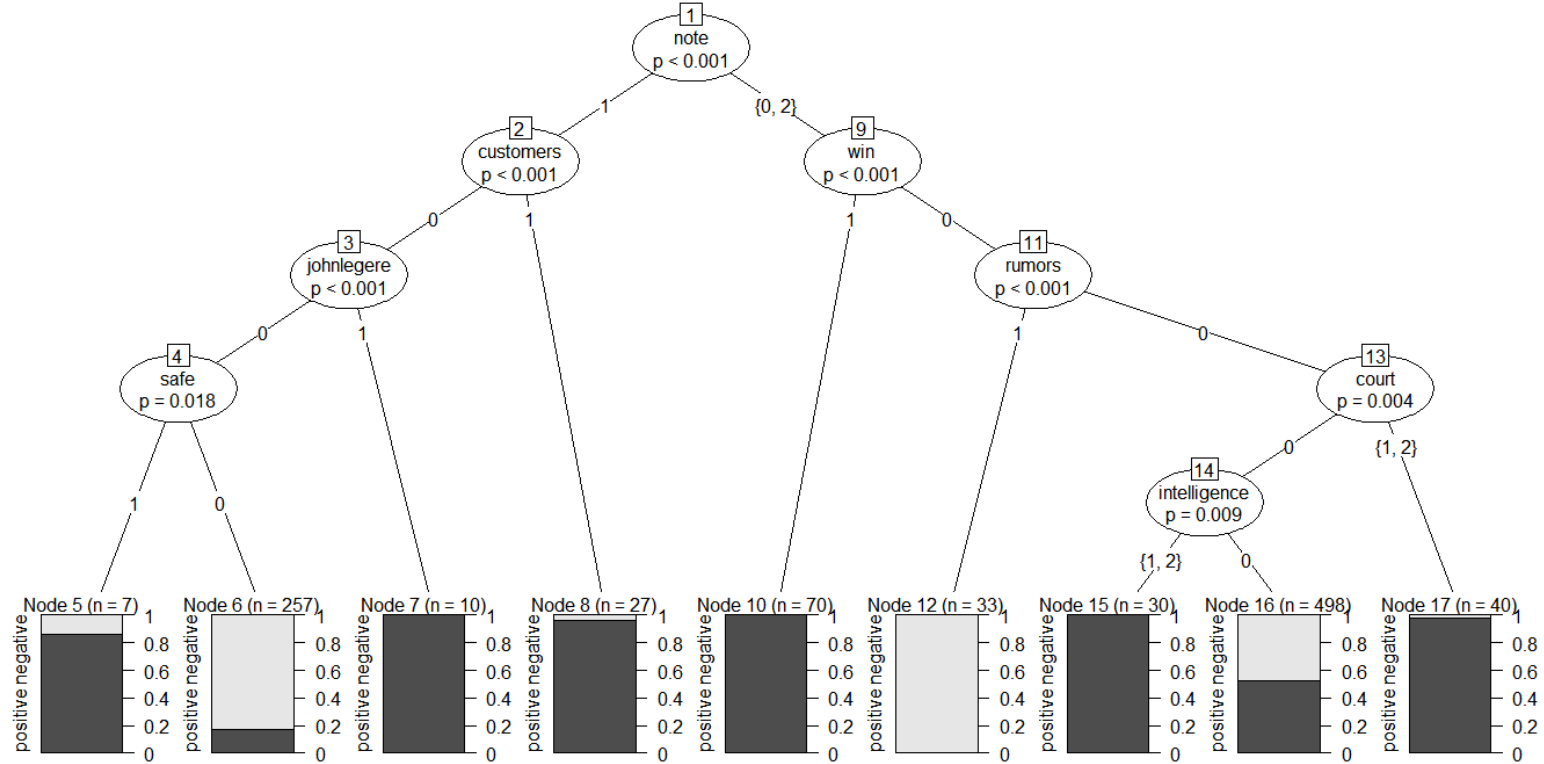
Árvore de Decisão

```
137 predicted3 = predict(fit3, newdata=dados)
138 predicted5 = predict(fit5, newdata=dados)
139 predicted10 = predict(fit10, newdata=dados)
140 predicted = predict(fit, newdata=dados)
141
142 tab3 <- table(dados[-id_train,1], predicted3[-id_train])
143 (tab3[1,1]+tab3[2,2])/sum(tab3)
144 tab5 <- table(dados[-id_train,1], predicted5[-id_train])
145 (tab5[1,1]+tab5[2,2])/sum(tab5)
146 tab10 <- table(dados[-id_train,1], predicted10[-id_train])
147 (tab10[1,1]+tab10[2,2])/sum(tab10)
148 tab <- table(dados[-id_train,1], predicted[-id_train])
149 (tab[1,1]+tab[2,2])/sum(tab)
```

Árvore de Decisão

```
> tab <- table(dados[-id_train,1], predicted[-id_train])
> (tab[1,1]+tab[2,2])/sum(tab)
[1] 0.744856
> plot(fit5)
> tab3 <- table(dados[-id_train,1], predicted3[-id_train])
> (tab3[1,1]+tab3[2,2])/sum(tab3)
[1] 0.744856
> tab5 <- table(dados[-id_train,1], predicted5[-id_train])
> (tab5[1,1]+tab5[2,2])/sum(tab5)
[1] 0.7530864
> tab10 <- table(dados[-id_train,1], predicted10[-id_train])
> (tab10[1,1]+tab10[2,2])/sum(tab10)
[1] 0.744856
> tab <- table(dados[-id_train,1], predicted[-id_train])
> (tab[1,1]+tab[2,2])/sum(tab)
[1] 0.744856
```

Árvore de Decisão



Referências Bibliográficas

- Hastie, T., Tibshirani, R. & Friedman, J.H. (2001) “The Elements of Statistical Learning”
- Bishop, C.M. (2007) “Pattern Recognition and Machine Learning”
- Mitchell, T.M. (1997) “Machine Learning”
- Abu-Mostafa, Y., Magdon-Ismail, M., Lin, H.T (2012) “Learning from data”
- Theodoridis, S., Koutroumbas, K., (2008) “Pattern Recognition”
- Kuhn, M., Johnson, K., (2013) “Applied Predictive Modeling”