DATASCI 207

Nedelina Teneva, PhD
nteneva@berkeley.edu

School of Information, UC Berkeley

Announcements

Week 6-8: decide on a project team and project dataset!

Async Quizz

- With n features and m training examples, the shape of the gradient should be?
- Linear regression requires numeric inputs.
- A categorical feature with 10 possible values produces a one-hot representation with 10 features.
- Z-Score scaling can significantly improve training convergence speed.
- MAE for car price prediction could be improved simply by clipping negative predictions to 0.

Feature Engineering

Async Quizz

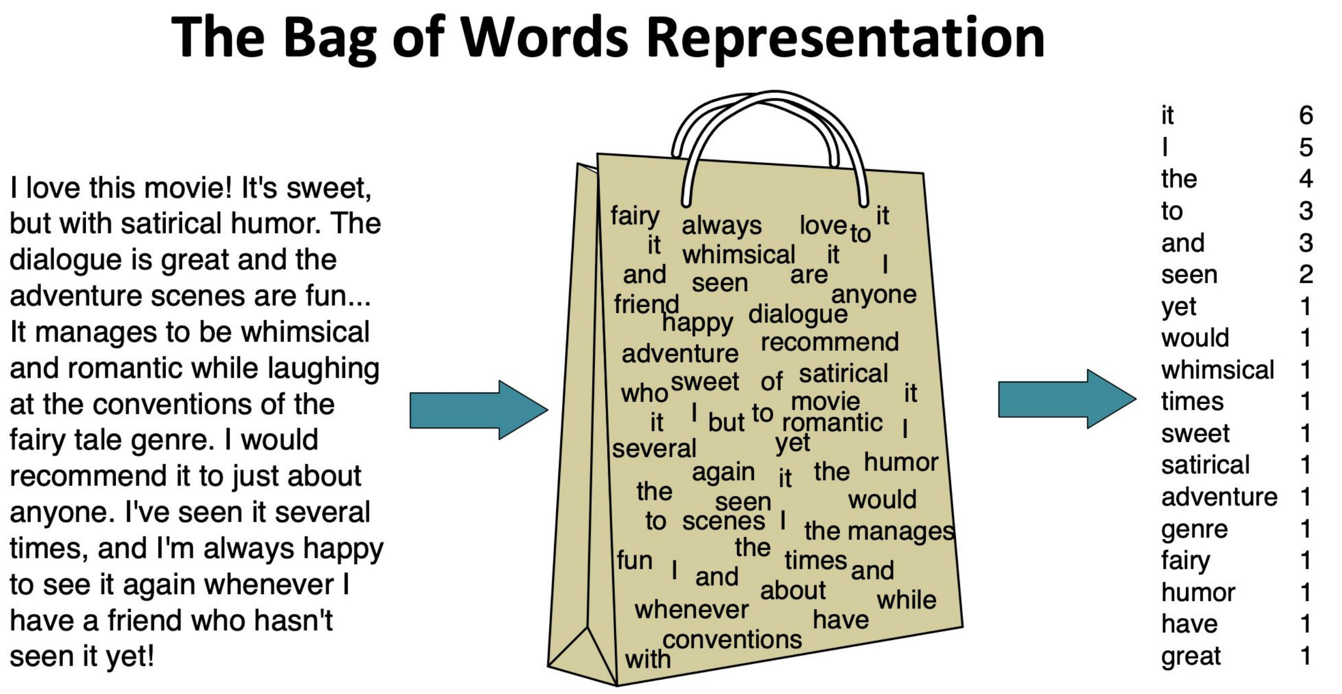
- With n features and m training examples, the shape of the gradient should be? (n,1)
- Linear regression requires numeric inputs. True
- A categorical feature with 10 possible values produces a one-hot representation with 10 features. True
- Z-Score scaling can significantly improve training convergence speed. True
- Mean Absolute Error (MAE) for car price prediction could be improved simply by clipping negative predictions to 0. True
  - y_i is the prediction; x_i is the true val

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n}$$

Natural Language Processing Example

- Sentiment Classification: is a review/text snippet positive or negative?
- Text classification into categories/topics/genres/:
    - Is an email spam? Is a job posting fake? Is a social media post fake?
    - Is an article about news, music, sports etc.?
    - Is a paper about biology, computer science, math?
    - Is a song jazz, classical, etc.?

Slide from: https://web.stanford.edu/~jurafsky/slp3/slides/7_NB.pdf



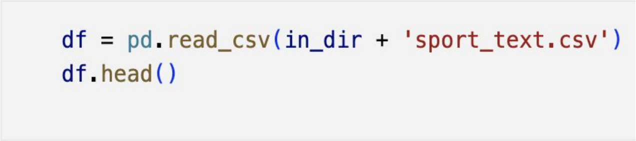
The Bag of Words Representation

I love this movie! It's sweet, but with satirical humor. The dialogue is great and the adventure scenes are fun... It manages to be whimsical and romantic while laughing at the conventions of the fairy tale genre. I would recommend it to just about anyone. I've seen it several times, and I'm always happy to see it again whenever I have a friend who hasn't seen it yet!

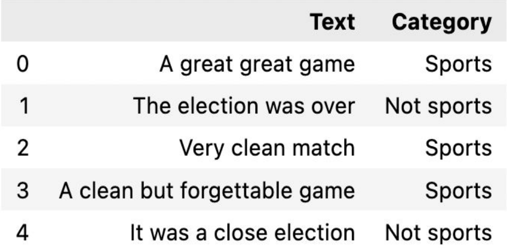
it	6
I	5
the	4
to	3
and	3
seen	2
yet	1
would	1
whimsical	1
times	1
sweet	1
satirical	1
adventure	1
genre	1
fairy	1
humor	1
have	1
great	1

Text Classification and Naive Bayes

- P(w_1, w_2, …, w_n | c) = P (w_1|c)P(w_2|c)....P(w_n|c)
- Bag of words assumption: the word order doesn’t matter!
- NB make a conditional independence assumption: the feature probabilities are independent given the class c.



```
df = pd.read_csv(in_dir + 'sport_text.csv')
df.head()
```



	Text	Category
0	A great great game	Sports
1	The election was over	Not sports
2	Very clean match	Sports
3	A clean but forgettable game	Sports
4	It was a close election	Not sports

Tools: Bayes' Theorem! useful when working with conditional probabilities.

$$P(Sports|a\,very\,close\,game) = \frac{P(a\,very\,close\,game|Sports) \times P(Sports)}{P(a\,very\,close\,game)}$$

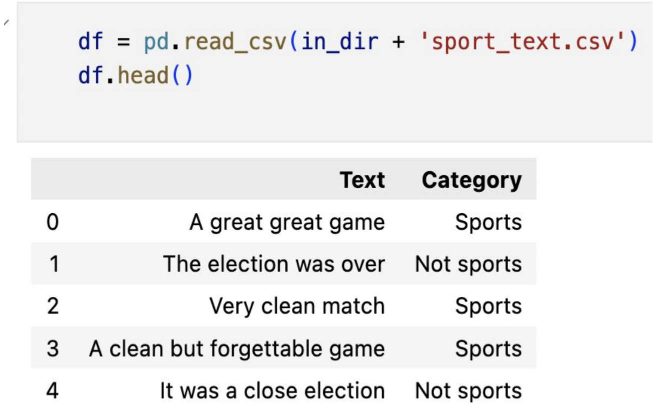
$$P(Not\,Sports|a\,very\,close\,game) = \frac{P(a\,very\,close\,game|Not\,Sports) \times P(Not\,Sports)}{P(a\,very\,close\,game)}$$

$$P(a\,very\,close\,game|Not\,sports) = P(a|Not\,sports) \times P(very|Not\,sports) \times P(close|Not\,sports) \times P(game|Not\,sports)$$

$$P(a\,very\,close\,game|Sports) = P(a|Sports) \times P(very|Sports) \times P(close|Sports) \times P(game|Sports)$$

What if we have a new string with unknown words?

https://github.com/MIDS-W207/nteneva/blob/main/live_sessions_current/week3/NB_Multinomial_classifier.ipynb (+ data file sports_text.csv)



```
df = pd.read_csv(in_dir + 'sport_text.csv')
df.head()
```

	Text	Category
0	A great great game	Sports
1	The election was over	Not sports
2	Very clean match	Sports
3	A clean but forgettable game	Sports
4	It was a close election	Not sports

Soccer, a great game

Missing Features

- Discard observations with any missing values
  - Feasible only if small number of values are missing
- Force the algorithm to deal with missing values
- Impute missing values
  - Simplest case is using the mean or median

Pre processing

- Lowercasing
- Stop words removal
- Language check/identification
- **Stem**ming: remove prefixes/suffixes; leave the stem
  - Drive; Driver; Driving have the stem “drive”
- **Lemma**tization: uses linguistic context to get the root lemma
  - “Better” reduced to the lemma “good”
  - Details: see https://nlp.stanford.edu/IR-book/html/htmledition/stemming-and-lemmatization-1.html
- Handling URLs, phone numbers, time stamps, encoding issues, context switching (HTML or code pieces in English text)

Linear Regression 2

- https://github.com/MIDS-W207/nteneva/blob/main/live_sessions_current/week3/Linear_Regression_II_new.ipynb