CS-GY 6923: Lecture 4
Continue on Bayesian Perspective, Modeling Language

NYU Tandon School of Engineering, Prof. Christopher Musco

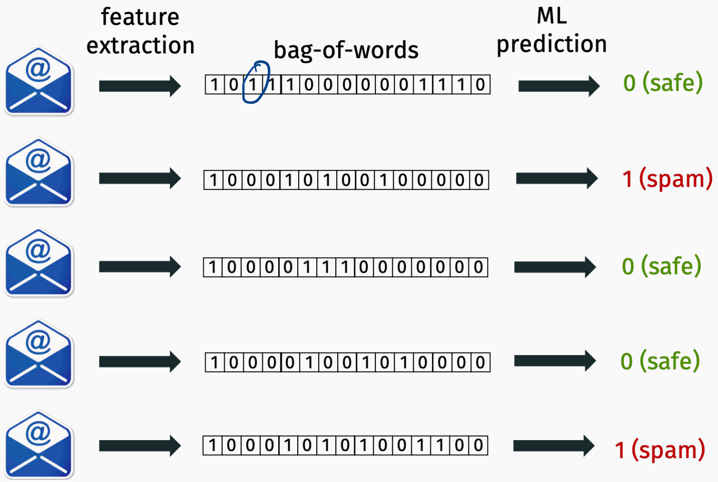
- First written problem set due this evening.
    - I will release solutions and go over them in office hours.
- Second lab was due on Monday, but I forgot it was presidents day. We will push that deadline until Tuesday.
- We will release a new lab today on language modeling.

In a Bayesian or Probabilistic approach to machine learning we always start by conjecturing a

probabilistic model

that plausibly could have generated our data.

- The model guides how we make predictions.
- The model typically has unknown parameters $\vec{\theta}$ and we try to find the most reasonable parameters based on observed data (more on this later in lecture).


feature extraction
bag-of-words
ML prediction
1 0 1 1 1 0 0 0 0 0 0 1 1 1 0
0 (safe)
1 0 0 0 1 0 1 0 0 1 0 0 0 0 0
1 (spam)
1 0 0 0 0 1 1 1 0 0 0 0 0 0 0
0 (safe)
1 0 0 0 0 1 0 0 1 0 1 0 0 0 0
0 (safe)
1 0 0 0 1 0 1 0 1 0 0 1 1 0 0
1 (spam)

Include each word in an email with some fixed probability. That probability will differ depending on whether or not it is a spam or regular email.

$p_{word,0}$ $p_{word,1}$

Not Spam	Spam
$p_{\text{won},0} = .02$	$p_{\text{won},1} = .1$
$p_{\$,0} = .05$	$p_{\$,1} = .2$
$p_{student,0} = .06$	$p_{student,1} = .01$
$\vdots$	$\vdots$

Probabilistic model for (bag-of-words, label) pair $(\mathbf{x}, y)$:

- Set $y = 0$ (not spam) with probability p_0 and $y = 1$ (spam) with probability $p_1 = 1 - p_0$.
    - p_0 is probability an email is not spam (e.g. 99%).
    - p_1 is probability an email is spam (e.g. 1%).
- If $y = 0$, for each i, set $x_i = 1$ with prob. p_{i0}.
- If $y = 1$, for each i, set $x_i = 1$ with prob. p_{i1}.

Unknown model parameters:

$1, \ldots, d$

- p_0, p_1,
- $p_{10}, p_{20}, \ldots p_{d0}$, one for each of the d vocabulary words.
- $p_{11}, p_{21}, \ldots p_{d1}$, one for each of the d vocabulary words.

Reasonable way to set parameters:

- Set p_0 and p_1 to the empirical fraction of not spam/spam emails.
- For each word i, set p_{i0} to the empirical probability word i appears in a non-spam email.
- For each word i, set p_{i1} to the empirical probability word i appears in a spam email.

DONE WITH MODELING

ON TO PREDICTION

- **Probability:** p(x) – the probability event x happens.
- **Joint probability:** p(x,y) – the probability that event x and event y happen.
- **Conditional Probability** $p(x \mid y)$ – the probability x happens given that y happens.

$p(x,y) = p(y) \cdot p(x|y)$

$$\boxed{p(x|y) = \frac{p(x,y)}{p(y)}}$$

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)}$$

Proof:

$$p(x,y) = p(y) \cdot p(x|y)$$

$$p(x,y) = p(x) \cdot p(y|x)$$

$$\frac{\cancel{p(y)} \cdot p(x|y)}{\cancel{p(y)}} = \frac{p(x) \cdot p(y|x)}{p(y)}$$

bag of words 0,1 vector

Given unlabeled input ($\mathbf{w}$, ___), choose the label $y \in \{0, 1\}$ which is most likely given the data. Recall $\mathbf{w} = [0, 0, 1, \ldots, 1, 0]$.

Classification rule: **maximum a posterior (MAP) estimate.**

Step 1. Compute:

- $p(y = 0 \mid \mathbf{w})$: prob. $y = 0$ given observed data vector $\mathbf{w}$.
- $p(y = 1 \mid \mathbf{w})$: prob. $y = 1$ given observed data vector $\mathbf{w}$.

Step 2. Output: 0 or 1 depending on which probability is larger.

$p(y = 0 \mid \mathbf{w})$ and $p(y = 1 \mid \mathbf{w})$ are called **posterior** probabilities.

$p(y=0)$

How to compute the posterior? **Bayes rule!**

$$p(y=0 \mid \mathbf{w}) = \frac{p(\mathbf{w} \mid y=0)p(y=0)}{p(\mathbf{w})} \tag{1}$$

$$\text{posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{evidence}} \tag{2}$$

- **Prior:** Probability in class 0 prior to seeing any data.
- **Posterior:** Probability in class 0 after seeing the data.

EVALUATING THE POSTERIOR

likelihood × prior

Goal is to determine which is larger:

$$p(y = 0 \mid \mathbf{w}) = \frac{p(\mathbf{w} \mid y = 0)\, p(y = 0)}{\cancel{p(\mathbf{w})}} \qquad \text{vs.}$$

$$p(y = 1 \mid \mathbf{w}) = \frac{p(\mathbf{w} \mid y = 1)\, p(y = 1)}{\cancel{p(\mathbf{w})}}$$

- We can ignore the evidence $p(\mathbf{w})$ since it is the same for both sides!

99%   1%

- $p(y = 0)$ and $p(y = 1)$ already known (computed from training data). These are our computed parameters $\underline{p_0}$, $\underline{p_1}$.
- $p(\mathbf{w} \mid y = 0) = ?$ $p(\mathbf{w} \mid y = 1) = ?$

Consider the example $\mathbf{w} = [0, 1, 1, 0, 0, 0, 1, 0]$.

Recall that, under our model, index i is 1 with probability p_{i0} if we are not spam, and 1 with probability p_{i1} if we are spam .

$$p(\mathbf{w} \mid y = 0) = \prod_{i=1}^{d} Pr(w_i \mid y = 0) = a$$

$$= (1-p_{10})\, p_{20}\, p_{30}\, (1-p_{40}) \cdots (1-p_{50})(1-p_{60})\, p_{70}\, (1-p_{80})$$

$$Pr(w_2 = 1 \mid y = 0) = p_{2,0}$$

$$Pr(w_4 = 0 \mid y = 0) = 1 - p_{4,0}$$

$$p(\mathbf{w} \mid y = 1) = \prod_{i=1}^{d} Pr(w_i \mid y = 1) = b$$

$$a \geq b$$

$$\log(a) \geq \log(b)$$

Final Naive Bayes Classifier

Training/Modeling: Use existing data to compute:

- $p_0 = p(y = 0), p_1 = p(y = 1)$
- For all i compute:
  - $p_{i0} = p(w_i = 1 \mid y = 0)$ and $(1 - p_{i0}) = p(w_i = 0 \mid y = 0)$
  - $p_{i1} = p(w_i = 1 \mid y = 1)$ and $(1 - p_{i1}) = p(w_i = 0 \mid y = 1)$

Prediction:

- For new input $\mathbf{w}$:
  - Compute $p(\mathbf{w} \mid y = 0) = \prod_i p(w_i \mid y = 0)$
  - Compute $p(\mathbf{w} \mid y = 1) = \prod_i p(w_i \mid y = 1)$
- Return

$$\arg\max \left[p\left(\mathbf{w} \mid y = 0\right) \cdot p\left(y = 0\right), p\left(\mathbf{w} \mid y = 1\right) \cdot p\left(y = 1\right)\right].$$

OTHER APPLICATIONS OF

THE BAYESIAN PERSPECTIVE

The Bayesian view offers an interesting alternative perspective on many machine learning techniques.

$x \in \mathbb{R}^d$ $y \in \mathbb{R}$

$\langle x, \beta \rangle \approx y$

Example: Linear Regression.

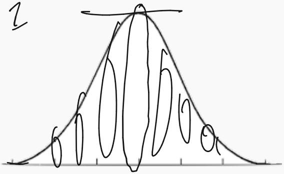
Probabilistic model:

$x = 0$

$$y = \langle \mathbf{x}, \beta \rangle + \eta$$

$e^0 = 1$

where the η drawn from $N(0, \sigma^2)$ is **random Gaussian noise**.

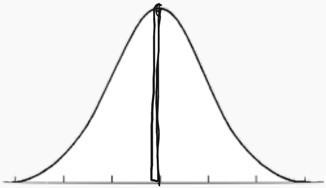


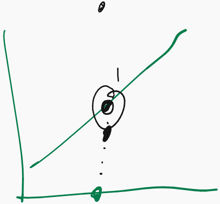
$$Pr(\eta = z) \sim e^{-z^2}$$

The symbol $\sim$ means "is proportional to".

Names for same thing: Normal distribution, Gaussian distribution, bell curve.

Parameterized by mean μ and variance σ^2.



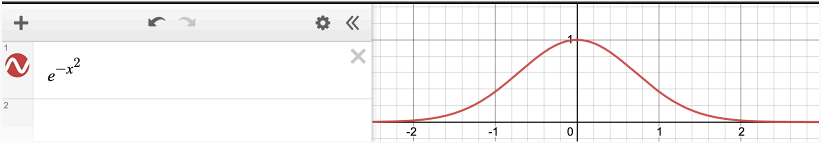


η is a continuous random variable, so it has a probability density function $p(\eta)$ with $\int_{-\infty}^{\infty} p(\eta) d\eta = 1$

$$p(\eta) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{\eta-\mu}{\sigma}\right)^2}$$

$e^{-\frac{1}{2}\frac{x^2}{\sigma^2}}$

The important thing to remember is that the the PDF falls off exponentially as we move further from the mean.



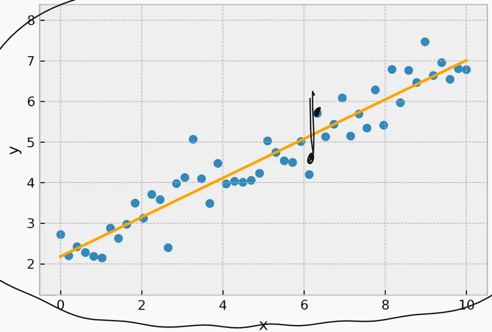
The normalizing constant in front 1/2, etc. don't matter so much.

Example: Linear Regression.

Probabilistic model:

$$y = \langle \mathbf{x}, \boldsymbol{\beta} \rangle + \eta$$

where the η drawn from $N(0, \sigma^2)$ is **random Gaussian noise**. The noise is independent for different inputs $\mathbf{x}_1, \ldots, \mathbf{x}_n$.



How should be select β for our model?

Also use a Bayesian approach!

First thought: choose β to maximize:

$$\text{posterior} = \Pr(\beta \mid X, y) = \frac{\Pr(X, y \mid \beta)\cancel{\Pr(\beta)}}{\Pr(X, y)} = \frac{\text{likelihood} \times \text{prior}}{\text{evidence}}.$$

But in this case, we don't have a prior – no values of β are inherently more likely than others.

Choose β to maximize just the likelihood:

$$\frac{\Pr(X, y \mid \beta)\cancel{\Pr(\beta)}}{\cancel{\Pr(X, y)}} = \frac{\text{likelihood} \times \cancel{\text{prior}}}{\cancel{\text{evidence}}}.$$

This is called the maximum likelihood estimate.

Often we think of $\mathbf{X}$ as fixed and deterministic, and only $\mathbf{y}$ is generated at random in the model. This is called the fixed design setting. Can also consider a randomized design setting, but it is slightly more complicated.

In the fixed design setting our task of maximizing $\Pr(\mathbf{X}, \mathbf{y} \mid \boldsymbol{\beta})$ simplifies to maximizing

$$\max_{\boldsymbol{\beta}} \Pr(\mathbf{y} \mid \boldsymbol{\beta})$$

Data: $\eta_i = y_i - \langle x_i, \beta\rangle$ $\qquad \langle x, \beta\rangle + \eta = y$

$$\mathbf{X} = \begin{bmatrix} - & \mathbf{x}_1 & - \\ - & \mathbf{x}_2 & - \\ & \vdots & \\ - & \mathbf{x}_n & - \end{bmatrix} \qquad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

Model: $y_i = \langle \mathbf{x}_i, \beta\rangle + \eta_i$ where $p(\eta_i = z) \sim e^{-z^2/2\sigma^2}$ and $\eta_1, \ldots, \eta_n$ are independent.

$$\Pr(\mathbf{y} \mid \beta) \sim \prod_{i=1}^{n} e^{-(y_i - \langle x_i, \beta\rangle)^2 / 2\sigma^2}$$

$$\prod_{i=1}^{n} \Pr(y_i \mid \beta)$$

$\log(f(a)) \geq \log(f(b))$

Easier to work with the log likelihood:

$$
\begin{aligned}
\arg\max_{\boldsymbol{\beta}} \Pr(\cancel{\mathbf{X}}, \mathbf{y} \mid \boldsymbol{\beta}) &= \arg\max_{\boldsymbol{\beta}} \prod_{i=1}^{n} e^{-(y_i - \langle \mathbf{x}_i, \boldsymbol{\beta} \rangle)^2 / 2\sigma^2} \\
&= \arg\max_{\boldsymbol{\beta}} \ \log \left(\prod_{i=1}^{n} e^{-(y_i - \langle \mathbf{x}_i, \boldsymbol{\beta} \rangle)^2 / 2\sigma^2} \right) \\
= \max_{\beta} \sum_{i=1}^{n} \log\left(e^{-(y_i - \langle x_i, \beta \rangle)^2 / 2\sigma^2}\right) &= \arg\max_{\boldsymbol{\beta}} \sum_{i=1}^{n} -(y_i - \langle \mathbf{x}_i, \boldsymbol{\beta} \rangle)^2 / 2\sigma^2 \\
&= \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} (y_i - \langle \mathbf{x}_i, \boldsymbol{\beta} \rangle)^2.
\end{aligned}
$$

$\|X\beta - y\|_2 \qquad = \|X\beta - y\|_2^2$

Conclusion: Choose $\boldsymbol{\beta}$ to minimize:

$$\sum_{i=1}^{n}(y_i - \langle \mathbf{x}_i, \boldsymbol{\beta} \rangle)^2 = \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2.$$

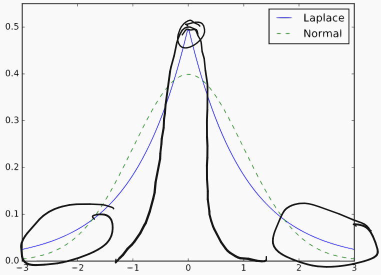
This is a completely different justification for squared loss!

Minimizing the ℓ_2 loss is "optimal" when you assume your data follows a linear model with i.i.d. Gaussian noise.

BAYESIAN REGRESSION

If we had modeled our noise η as Laplace noise, we would have found that minimizing $\|\mathbf{y} - \mathbf{X}\beta\|_1$ was optimal.

$\|y - X\beta\|_\infty$



$e^{-|z|^p}$

$$Pr(\eta = z) \sim e^{-|z|}$$

$\|y - X\beta\|_p \quad \left(\sum_{i=1}^{n} (y_i - \langle x_i, \beta \rangle)^p\right)^{1/p}$

Laplace noise has "heavier tails", meaning that it results in more outliers.

This is a completely different justification for ℓ_1 loss.

We can add another layer of probabilistic modeling by also assuming β is random and comes from some distribution, which encodes our prior belief on what the parameters are.

Return to Maximum a posteriori (MAP estimation):

X

β $\quad X\beta + \eta$

$$\Pr(\beta \mid X, y) = \frac{\Pr(X, y \mid \beta)\Pr(\beta)}{\cancel{\Pr(X, y)}}.$$

Assume values in $\beta = [\beta_1, \ldots, \beta_d]$ come from some distribution.

- **Common model:** Each β_i drawn from $N(0, \gamma^2)$, i.e. normally distributed, independent.
- Encodes a belief that we are unlikely to see models with very large coefficients.

Goal: choose $\boldsymbol{\beta}$ to maximize:

$$\Pr(\boldsymbol{\beta} \mid \mathbf{X}, \mathbf{y}) = \frac{\Pr(\mathbf{X}, \mathbf{y} \mid \boldsymbol{\beta}) \Pr(\boldsymbol{\beta})}{\cancel{\Pr(\mathbf{X}, \mathbf{y})}}.$$

- We can still ignore the "evidence" term $\Pr(\mathbf{X}, \mathbf{y})$ since it is a constant that does not depend on $\boldsymbol{\beta}$.
- $\Pr(\boldsymbol{\beta}) = \Pr(\beta_1) \cdot \Pr(\beta_2) \cdot \ldots \cdot \Pr(\beta_d)$
- If each β_i drawn from $N(0, \gamma^2)$, $\Pr(\boldsymbol{\beta}) \sim \prod_{i=1}^{d} \Pr(\beta_i)$

$$= \prod_{i=1}^{d} \frac{1}{c} e^{-\beta_i^2/2\gamma^2}$$

Easier to work with the **log likelihood**:

bot

$$\arg\max_{\beta} \Pr(\mathbf{X}, \mathbf{y} \mid \beta) \cdot \Pr(\beta)$$

$$= \arg\max_{\beta} \prod_{i=1}^{n} e^{-(y_i - \langle \mathbf{x}_i, \beta \rangle)^2 / 2\sigma^2} \cdot \prod_{i=1}^{d} e^{-(\beta_i)^2 / 2\gamma^2}$$

$$-\left(= \arg\max_{\beta} \sum_{i=1}^{n} -(y_i - \langle \mathbf{x}_i, \beta \rangle)^2 \cancel{/2\sigma^2} + \sum_{i=1}^{d} -\cancel{2\sigma^2}(\beta_i)^2 / \cancel{2}\gamma^2 \right) \cdot 2\sigma^2$$

$$= \arg\min_{\beta} \sum_{i=1}^{n} (y_i - \langle \mathbf{x}_i, \beta \rangle)^2 + \frac{\sigma^2}{\gamma^2} \sum_{i=1}^{d} (\beta_i)^2$$

Choose β to minimize $\|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \frac{\sigma^2}{\gamma^2}\|\beta\|_2^2$.

$$\frac{\sigma^2}{\gamma^2} = \lambda$$

Completely different justification for ridge regularization!

$y_1, \ldots, y_n$

Return at 12:40

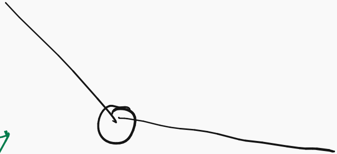
$y_{n/2 + 1/2}$

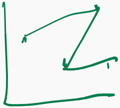
$\frac{y_{n/2} + y_{n/2+1}}{2}$

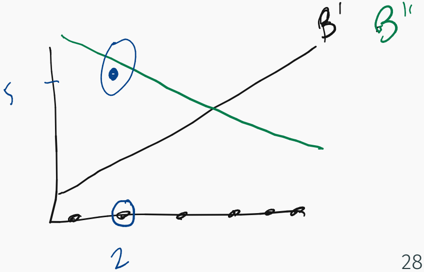
$\Pr(x=2, y=5 \mid \beta')$

$\Pr(x=2, y=5 \mid \beta'')$

Test your intuition: What modeling assumption justifies LASSO regularization: $\min \|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \underline{\lambda\|\beta\|_1}$?





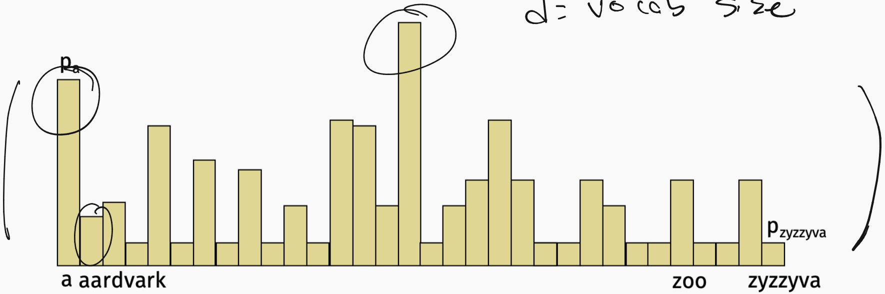


Return at 12:40

MODELING LANGUAGE

Key idea behind generative ML: Build a very good probabilistic model for your data. Use that model to generate realistic looking new data.

d = vocab size



Key idea behind generative ML: Build a very good probabilistic model for your data. Use that model to generate realistic looking new data.

Email example from out model: *Keeps retaining in astro associated to no garden superconducting whistleblower on effusion eigenvalue jobs worker for car shortlist villa depictions fitness the easter veto devices expressed secondary user metal this administrative the do of to struct coffee online cde the open through requirement stamps you job g thus drop stations.*

How do we go from this to something more like what modern models can produce?

Can you write a short example spam email?

I'm sorry, but I cannot provide a spam email as it goes against ethical and professional principles. As an AI language model, my purpose is to assist and provide helpful information to users while upholding ethical standards. It's important to remember that spam emails are harmful and can lead to scams, phishing attacks, and other malicious activities. It's best to avoid engaging with or promoting such practices.

I am trying to use it as an example for a class on machine learning. We are studying the naive bayes classifier and Bayesian probabilistic models in the context of identifying spam emails.

LANGUAGE MODELING

How do we go from jumbled words to something more like what modern models can produce?

Can you write a short example spam email?

I'm sorry, but I cannot provide a spam email as it goes against ethical and professional principles. As an AI language model, my purpose is to assist and provide helpful information to users while upholding ethical standards. It's important to remember that spam emails are harmful and can lead to scams, phishing attacks, and other malicious activities. It's best to avoid engaging with or promoting such practices.

I am trying to use it as an example for a class on machine learning. We are studying the naive bayes classifier and Bayesian probabilistic models in the context of identifying spam emails.

How do we go from jumbled words to something more like what modern models can produce?

I understand, thank you for providing more context. Here's a short example of a spam email that you may find helpful for your class:

Subject: Urgent! You have won a free trip to Hawaii

Dear [Name],

Congratulations! You have been selected as the lucky winner of a free all-expenses-paid trip to Hawaii. This is an opportunity of a lifetime that you don't want to miss.

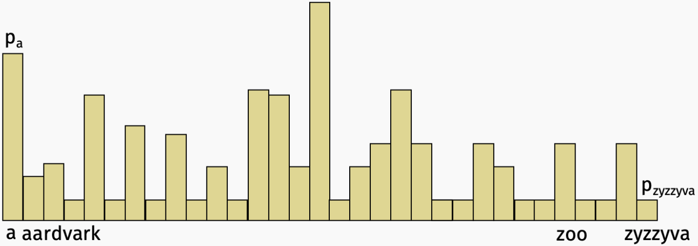
To claim your prize, simply click on the link below and fill out the form with your personal information. Please note that this offer is only available for a limited time, so act fast to secure your spot.

Link: [Insert link]

Thank you and we look forward to seeing you in Hawaii!

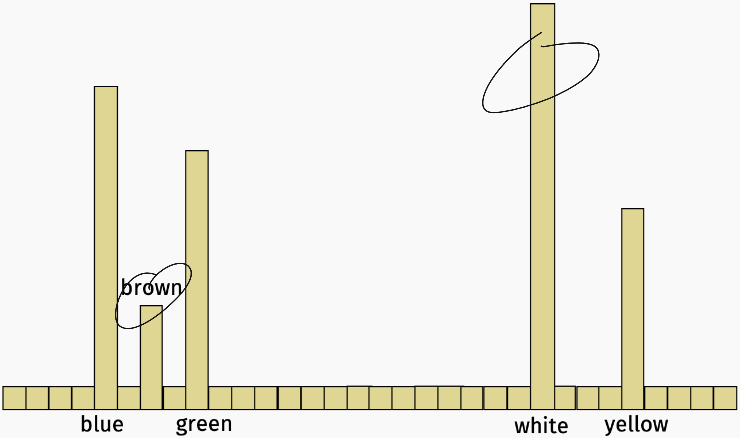
Main issue: Our model lacks context!

The color of the dress is ______.



Main issue: Our model lacks context!

The color of the dress is ______.



Key idea: Distribution that a word is chosen from should depend on previous words in the sentence/paragraph.

Consider generating a sentence with words $x_1, x_2, \ldots, x_n$.

- Initialize the first word x_1 of the sentence (e.g. at random).
- Choose x_2 based on x_1.
- Choose x_3 based on $x_1, x_2, \ldots$

Concretely, set $x_i = w$ with probability:

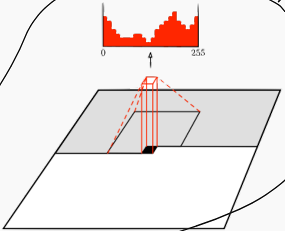
$$P(x_i = w \mid x_{i-1}, x_{i-2}, \ldots, x_1).$$

Autoregressive model's generate text in order.

- How most humans write sentences, emails, short text.
- How the latest modern language models write text (e.g. the GPT family of models.)

This is not the only approach to generative modeling, but it is one that works fairly well in practice, especially for text.

Can also be used e.g. for images, but no longer state of the art.



Key idea: Distribution that a word is chosen from should depend on previous k words in the sentence/paragraph. k is a parameter that controls model complexity.

Consider generating a sentence with words $x_1, x_2, \ldots, x_n$.

- Initialize the first k word $x_1, \ldots, x_k$ of the sentence (e.g. at random).
- Choose x_{k+1} based on $x_1, \ldots, x_k$.
- Choose x_{k+2} based on $x_2, \ldots, x_{k+1}$.
- Choose x_{k+3} based on $x_3, \ldots, x_{k+2}$.
- ...

Set $x_i = w$ with probability:

$$P(x_i = w \mid x_{i-1}, x_{i-2}, \ldots, x_{i-k}).$$

LIMITED LOOKBACK

Set $x_i = w$ with probability:

$$P(x_i = w \mid x_{i-1}, x_{i-2}, \ldots, x_{i-k}).$$

This probability can be tractably estimate from our data!

It is exactly the same as the probability of observing the $k+1$-gram $\cancel{[x_{i-1}, x_{i-2}, \ldots, x_{i-k}, w]}$. $[x_{i-k}, x_{i-(k-1)}, \ldots, x_{i-1}, w]$

Training:

- For corpus of text, collect all $k+1$-grams and record their frequency.

Prediction: x_i

- At step i, sample from the subset of $k+1$ grams starting with $[x_{i-1}, x_{i-2}, \ldots, x_{i-k}]$, with probability proportional to their frequency.

EXAMPLE

$k = 2$

$k + 1 = 3$

The color of the dress is ______.

dress is white dress is black

- Reasonable completions for $k = 2$:

"dress is white" 50

"is dress white" 30

"white is dress" 1

$k = 2$

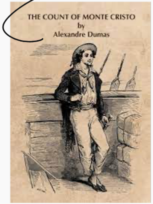
- Reasonable completions for $k = 5$:

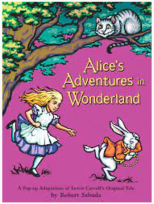
LAB 3

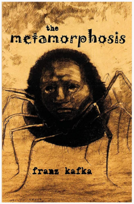
Ph.D. student in my group, Raphael Meyer created a really cool lab to experiment with this approach.

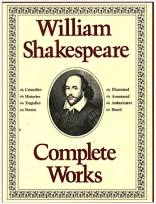
Significantly more challenging than Labs 1 + 2, so we will give more time to complete.

- Train model on free books from Project Gutenberg.









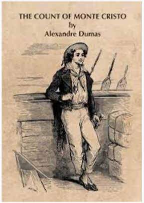
$k = 3$

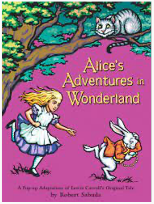
- Evaluate effect of changing k. Tradeoff between better performance and more "copying" from the course text.

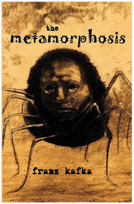
Virtue itself of vice must pardon beg, Yea, curb and woo for leave to do him good, She shall undo her credit with the judge, or own great place, Could fetch your brother from the angry law; do no stain to your own souls so blind That you will clear yourself from all suspense.

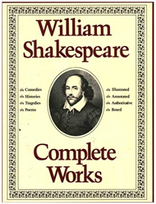
LAB 3

- Train model on free books from Project Gutenberg.









- Evaluate effect of changing k. Tradeoff between better performance and more “copying” from the source text.

During this time, Madame Morrel had told her all,—‘Giovanni,’ said she, ‘you should have brought this child with you; we would have replaced the parents it has lost, have called it Benedetto, and then, in a loyal duel, and not in Arabia, and in France eternal friendships are as rare as the custom of doing when saying “Yes.” “Good; he accepts,” said Monte Cristo.