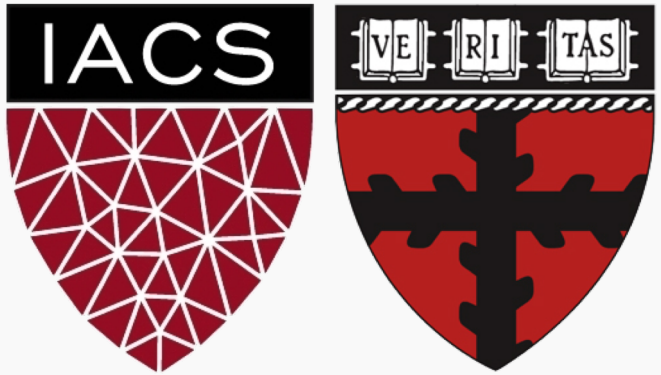
Probability Modeling of Linear Regression

CS109A Introduction to Data Science

Pavlos Protopapas, Kevin Rader and Chris Tanner



Lecture Outline

- Probability Review
  - Binomial Distribution
  - Normal Distribution
- Modeling Data with Probability Distributions
- Likelihood Theory
- Modeling Linear Regression Probabilistically
- `statmodels`

What is a random variable?

In the context of data, we often describe their potential **numeric** outcomes (before collecting the data) as random variables. That is:

Let's perform a survey of Harvard students and ask the question: do you primarily use a Mac (vs. PC vs. Linux/Ubuntu, etc.)?

Let X_1 be the observed response for the first person we are going to ask. Then X_1 can be thought of as a random variable. ($X_1 = 1$ implies 'Mac', $X_1 = 0$ implies anything else).

*Technically a random variable is a function that takes possible outcomes of random phenomenon (responses of 'Mac', 'PC', etc.) and maps them to numeric values.

What is a probability distribution?

A **probability distribution** is any function (formula, table, or graph) that assigns probabilities (or likelihoods) to all the possible outcomes of a random variable.

Typically they are written as a formula (called a probability mass function or probability density function, or as its cumulative distribution function).

In our 'Mac' example, we could define the probability distribution as a table:

x	$P(X = x)$
0	$1 - p$
1	p

Which could be summarized as the formula, for $x \in \{0,1\}$:

$$P(X = x) = p^x(1-p)^{1-x}$$

The goal of out study would be to estimate p.

Discrete vs. Continuous

There are two major types of random variables: discrete (can only take on specific values) and continuous (can take on any value within a range).

The probability distribution is defined differently for these two types:

A **probability mass function** (PMF) is a function that gives the probability of getting a specific value for a discrete random variable.

A **probability density function** (PDF) is a function that gives the relative likelihood of a specific value for a continuous random variable (that height of the curve).

*Note: probabilities for a continuous random variable can be represented as areas under the curve, and thus $P(X = x) = 0$ since there is no width.

The Binomial Distribution

Let *X* be a random variable that counts the number of successes in a fixed number of independent trials (*n*) with fixed probability of success (*p*) in each trial. Then is said to have a **binomial distribution**. This is often written as: $X \sim Binom(n, p)$, and *X* has probability mass function (PMF):

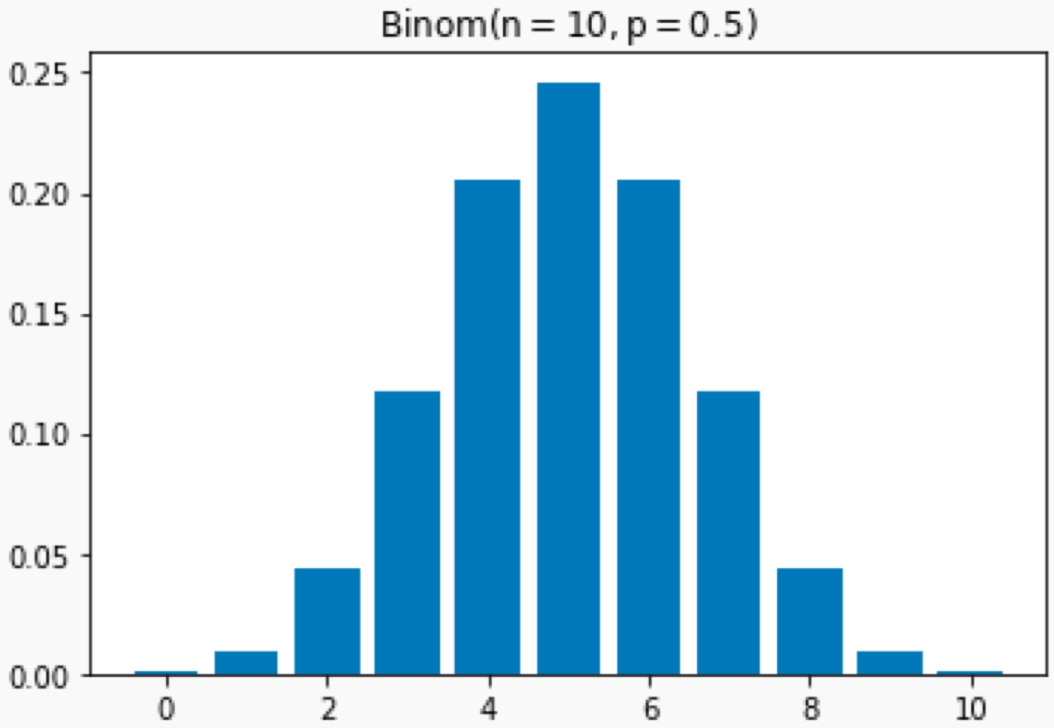
$$P(X = x) = \binom{n}{x} p^x (1-p)^{n-x}$$

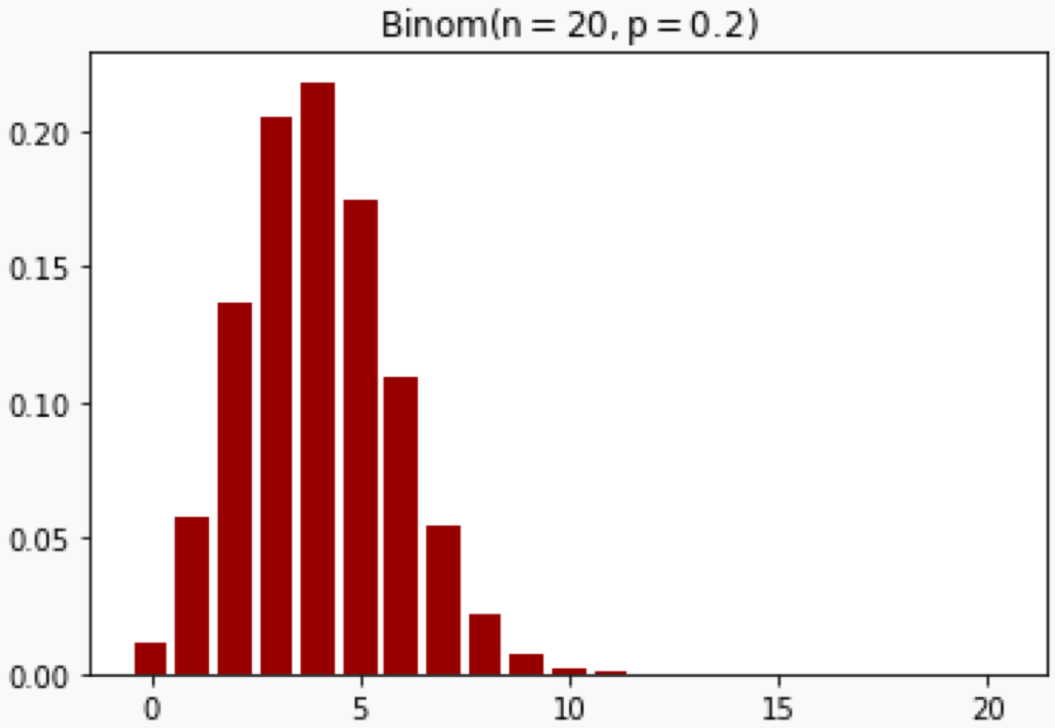
Think counting the number of heads when flipping a biased coin *n* times.

The binomial distribution is useful to describe polling data (proportion of people who will vote for Biden), survey data (will you take CS109B next year?), or any data that are binary!

The Bernoulli distribution is a special case when *n* = 1. This is the distribution that describes our `Mac` example.

Binomial Distribution Examples





A binomial distribution has mean np and standard deviation $\sqrt{np(1-p)}$.

The Normal Distribution

Let *X* be a **normally distributed** random variable. Then $X \sim N(\mu, \sigma^2)$, and *X* has probability distribution function (PDF):

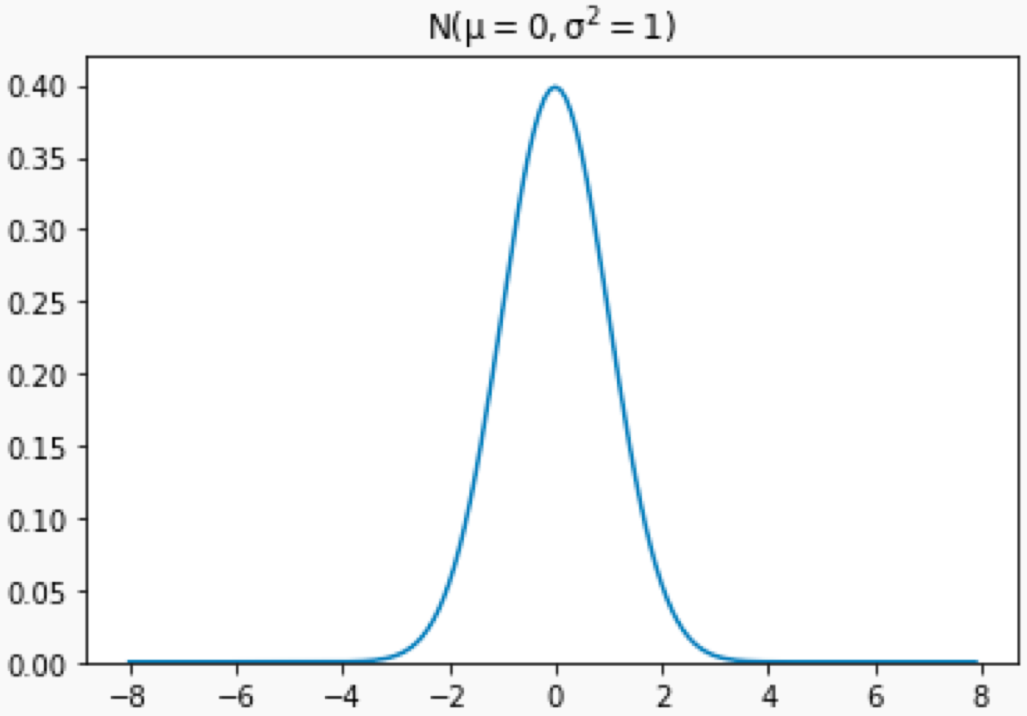
$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\left(\frac{x-\mu}{\sigma}\right)^2}$$

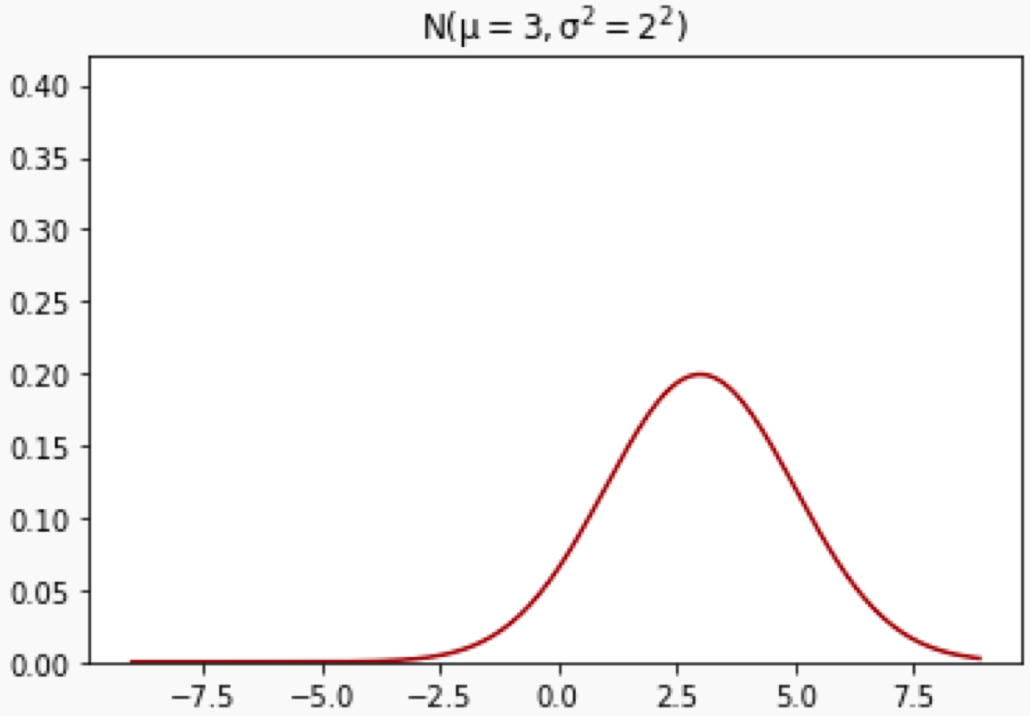
The normal distribution (sometimes called the Gaussian) is often referred to as the bell-shaped curve. But the normal distribution isn't the only one that is bell-shaped: *t* distributions are also bell shaped, for example.

The standard normal distribution is a special case: $Z \sim N(0,1)$.

Any normal random variable can be standardized using the formula $Z = \frac{X-\mu}{\sigma}$.

The Normal Distribution Examples





A normal distribution has mean μ and standard deviation σ.

Central Limit Theorem

Why is the normal distribution used so often? The **Central Limit Theorem**: random variables that are averages or sums of many other random variables will be approximately Normally distributed.

More specifically: if $X_1, X_2, \ldots, X_n$ are independent random variables (representing individual observations of data) with mean μ and standard deviation σ (not necessarily normal themselves), then the sample mean $\bar{X} = \frac{X_1 + X_2 + \cdots + X_n}{n}$ will have approximate distribution:

$$\bar{X} \sim N\left(\mu, \frac{\sigma^2}{n}\right)$$

Joint Distributions

What happens to these probability distributions (PMFs and PDFs) when there are multiple random variables (aka, multiple observations in a data set) involved?

Let $f(x_1, x_2, \ldots, x_n)$ be the **joint distribution** of n separate random variables. If they all come from the same generative marginal distribution, $f(x_i)$, and are independent, what is the resulting distribution?

$$f(x_1, x_2, \ldots, x_n) = f(x_1) \cdot f(x_2) \cdots f(x_n) = \prod_{i=1}^{n} f(x_i)$$

Modeling Data with Probability Distributions

The Probability of Data

In a typical probability problem (like in Stat 104 or 110), you would be told something like "20% of Harvard College students are collegiate athletes. What is the probability that there are 50 athletes in a random sample of 200 students from Harvard College?"

$$P(X = 50) = \binom{200}{50}(0.20)^{50}(0.80)^{150} = 0.0149$$

$$P(X \geq 50) = \sum_{50}^{200}\binom{200}{x}(0.20)^{x}(0.80)^{200-x} = 0.0494$$

An alternative question: what is more likely to occur: 50 athletes or 40 athletes in a sample of 200 students? How can we make the determination?

Inference: the inverse of probability

In the last problem, how did we know that the statement "20% of Harvard College students are collegiate athletes" is accurate? Where did this come from?

In most applications, the true population parameter (here, the proportion in all of Harvard College) is unknown. What we get to observe is the data, and we want to make a statement about the unknown parameter. So a more poignant question would be:

"There are 50 athletes in a random sample of 200 students from Harvard College. Is a binomial distribution with $p = 0.2$ or $p = 0.25$ more reasonable?"

This approach of using the data to make a statement about a parameter (in a statistical model) is called **inference**.

Likelihood Theory

The idea of likelihood

The **likelihood** approach to inference is based on exactly what was presented in the last slide: given observed values of data (summarized by specific sample statistics), what values of the model's parameters are likely?

It simply just flips a PDF or PMF on its head: instead of writing this function with the data (X) as the unknown, it uses the same function but uses the parameter(s) as the unknown(s). The **likelihood function**, $\mathcal{L}$, measures how well a model (and its set of parameters) describes the observed data

For a set of independent and normally distributed random variables, $X_i \sim N(\mu, \sigma^2)$:

$$\mathcal{L}(\mu, \sigma^2 | x_1, \ldots, x_n) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\left(\frac{x_i - \mu}{\sigma}\right)^2}$$

The log-likelihood

The likelihood function measures how well a model describes observed data. So it makes sense that we want a model (or set of parameters) that maximizes this function.

Likelihood function are typically products of many similar pieces, and products are difficult to maximize (both mathematically and numerically). Why?

So instead, the log of the likelihood function, called the **log-likelihood function**, ℓ, is used. For the Normal distribution model:

$$\ell\left(\mu, \sigma^2 | x_1, \ldots, x_n\right) = \ln\left(\prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\left(\frac{x_i-\mu}{\sigma}\right)^2}\right) = -\sum_{i=1}^{n} \sqrt{2\pi\sigma^2} - \sum_{i=1}^{n} \left(\frac{x_i-\mu}{\sigma}\right)^2$$

If the goal is optimization, why is transforming via the log function a good choice?

Maximizing the likelihood

In order to choose the best Normal distribution to describe a set of data, we should maximize the likelihood that chooses the best set of parameters given the data.

The **maximum likelihood estimates** for a statistical model are those that maximize the likelihood function given the observed data.

How do we do this mathematically? How could we do this computationally?

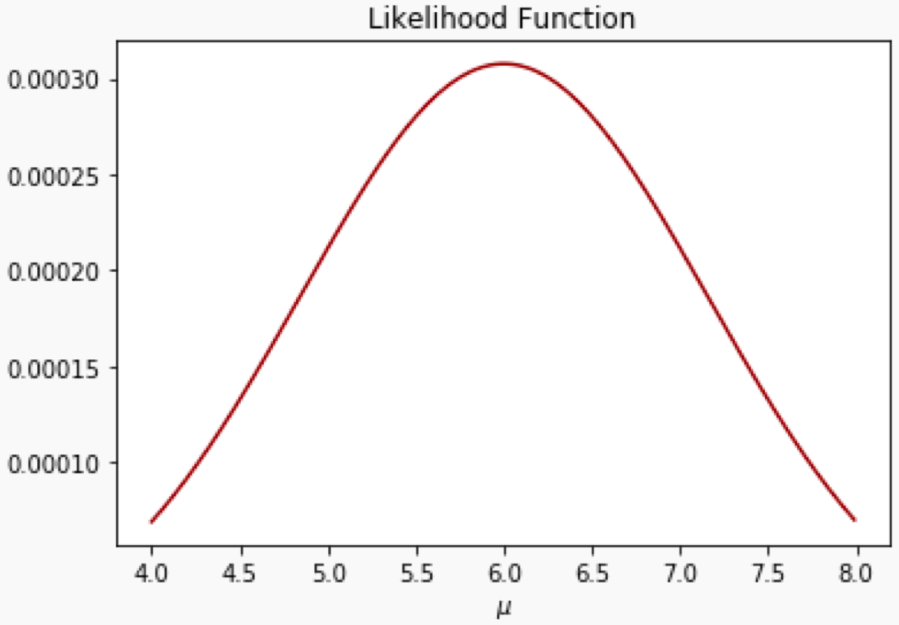
With Math: _____

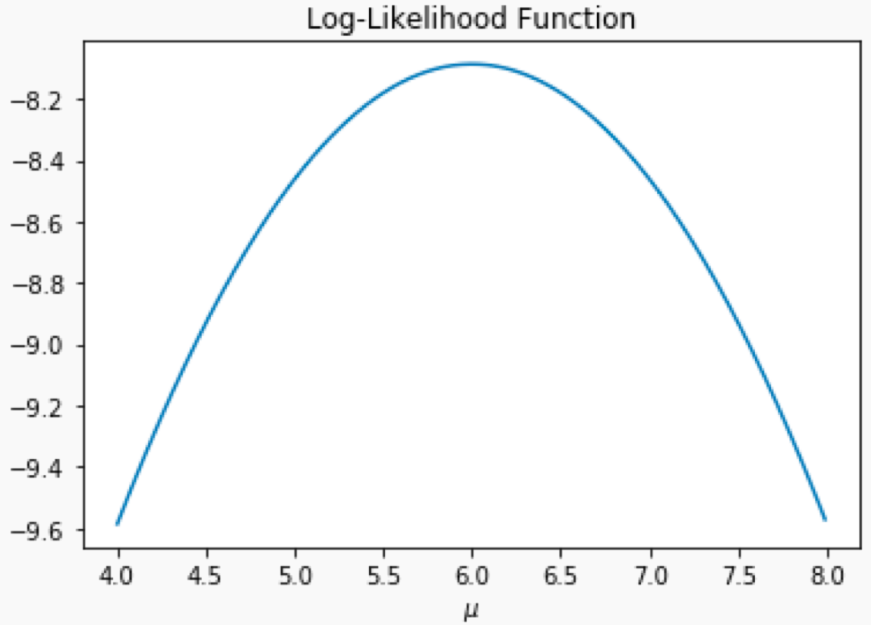
With Computers:_____

Likelihood function example

3 observations are collected [3, 5, 10] that are thought to come from a normal distribution with unknown mean, μ, but is known to have a variance of $\sigma^2 = 2^2$, (yes, this is contrived).

Let's plot the likelihood and log-likelihood functions:





Modeling Linear Regression Probabilistically

The Simple Linear Regression Model

We've defined the linear regression model to predict the *i*-th observation's response, Y_i, from a predictor, X_i, to be:

$$E(Y_i) = \beta_0 + \beta_1 X_i$$

This is often written instead as

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$$

The error term, ε_i, represents the distance the observation lies from the line in the vertical distance (direction of *Y*).

What's the difference between β_0, β_1 and $\hat{\beta}_0, \hat{\beta}_1$? What about Y_i and $\hat{Y}_i$?

The Probabilistic Regression Model

Let's rewrite the linear regression model with a probabilistic twist:

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i, \text{ where } \varepsilon_i \sim N(0, \sigma^2)$$

This regression model can be rewritten as:

$$Y_i | X_i \sim N(\beta_0 + \beta_1 X_i, \sigma^2)$$

This formulation allows us to write out the joint likelihood function for this probability model.

The Likelihood of Linear Regression

The joint likelihood function for this probability model becomes:

$$L(\beta_0, \beta_1 | \, \vec{y}, \vec{x}) = \prod_{i=1}^{n} f(y_i | x_i) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\left(\frac{y_i - (\beta_0 + \beta_1 x_i)}{\sigma}\right)^2}$$

Which leads to the log-likelihood:

$$l(\beta_0, \beta_1 | \, \vec{y}, \vec{x}) = -\sum_{i=1}^{n} \ln\left(\sqrt{2\pi\sigma^2}\right) - \sum_{i=1}^{n} \left(\frac{y_i - (\beta_0 + \beta_1 x_i)}{\sigma}\right)^2$$

What should we do with this log-likelihood?

What does maximizing this function lead to with regards to the best estimates of β_0, β_1?

Take home message

By taking a probabilistic approach to linear regression and assuming the residuals are normally distributed, we see that maximizing the likelihood to this model is equivalent to minimizing mean squared error!

So if we believe our residuals are normally distributed, then minimizing mean square error is a natural choice.

But by choosing this specific probability model, we get much more than simply motivation for our loss function. We get inferential

Checking the assumptions of this model:

The probabilistic model of linear regression leads to 4 main assumptions that can be checked with the data (the first 3 at least):

1. Linearity: relationships are linear and there is no clear non-linear pattern around the line (as evidenced by the residuals).
2. Normality: the residuals are normally distributed.
3. Constant Variance: the vertical spread of the residuals is constant everywhere along the line.
4. Independence: the observations are independent of each other.

Note: collinearity is not a violation of an assumption, but can certainly muck up the model.

Reminder: Estimates of the slope and intercept

Standard ordinary least squares (OLS) regression leads to explicit formulae for $\hat{\beta}_0$ and $\hat{\beta}_1$:

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$

Note: our probabilistic model states that the Y_i are normally distributed (conditional on X_i) and thus $\hat{\beta}_1$ and $\hat{\beta}_0$ will be normally distributed! We can leverage this to determine the sampling distribution of these estimates (and build hypothesis tests and confidence intervals*)!

*See lecture 9 for these approaches.

statsmodels

Fitting linear regression models in Python

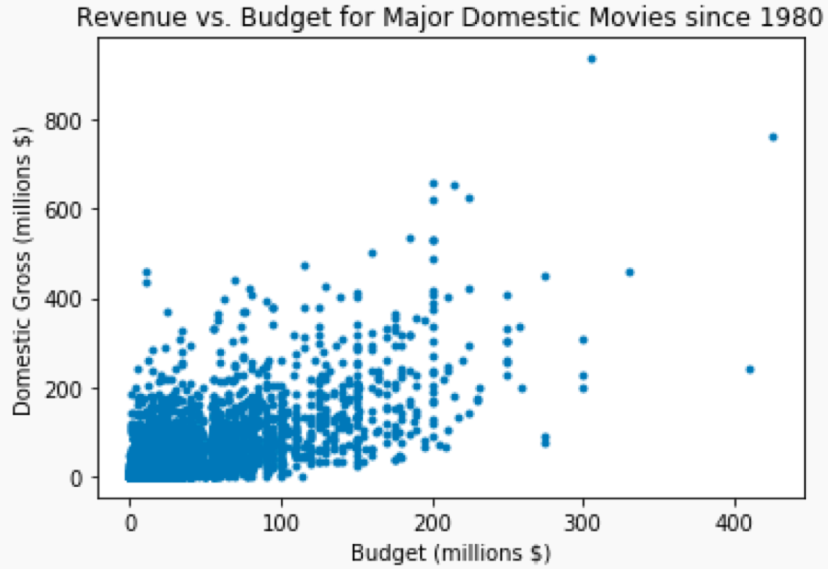
There are two packages used to fit linear regression models in python.

- **`sklearn`**: is great for getting estimates, doing predictions, integrating cross-validation, etc. Not so good for confidence intervals or hypothesis testing (it's machine learning, not statistics).

- **`statsmodels`**: is great for statistical inference (confidence intervals and hypothesis testing) but not as good at the other things

Which to use depends on what your goal of modeling is.

OLS in `statsmodels`



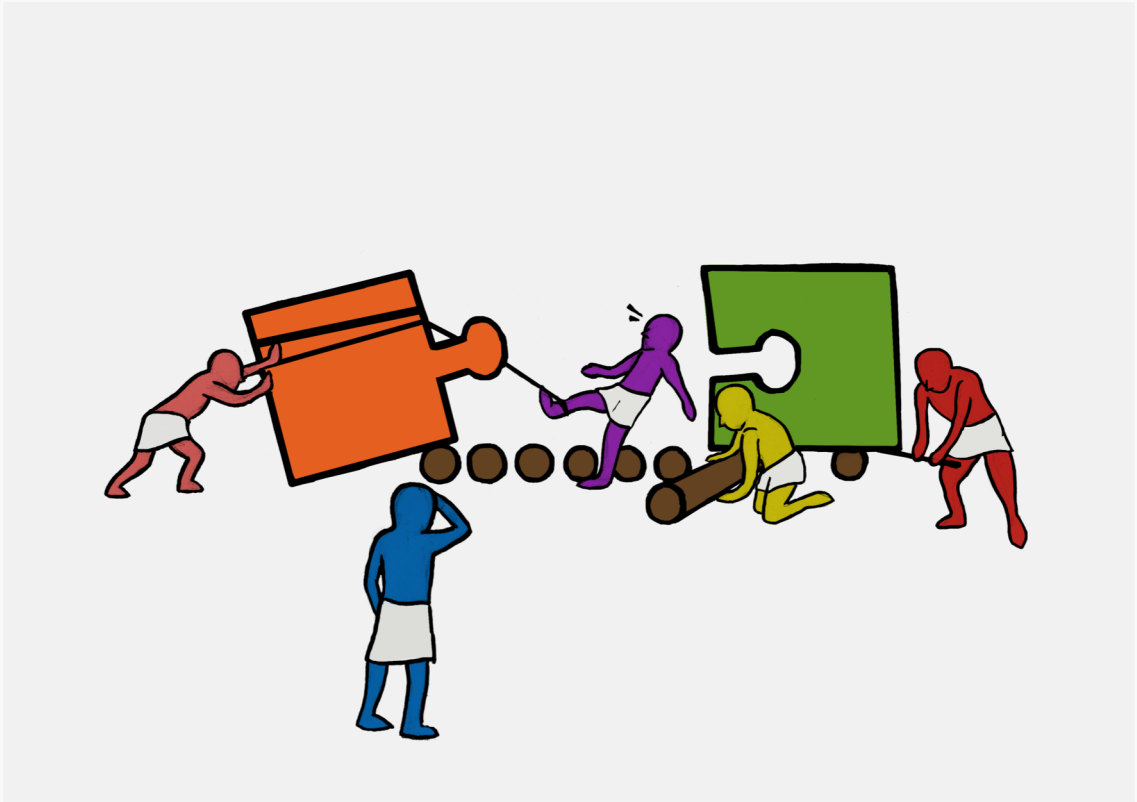

```
import statsmodels.api as sm
from sklearn.linear_model import LinearRegression

X = sm.add_constant(movies_data['budget'])
ols1 = OLS(movies_data['domestic'],X).fit()

regress = LinearRegression().fit(
    movies_data[['budget']],movies_data['domestic'])
print(regress.coef_,regress.intercept_)

ols1.summary()
```

```
[1.11222637] 7.282264927408129
```

Dep. Variable:	domestic	**R-squared:**	0.463
Model:	OLS	**Adj. R-squared:**	0.463
Method:	Least Squares	**F-statistic:**	4505.
Date:	Mon, 21 Sep 2020	**Prob (F-statistic):**	0.00
Time:	00:05:18	**Log-Likelihood:**	-27815.
No. Observations:	5222	**AIC:**	5.563e+04
Df Residuals:	5220	**BIC:**	5.565e+04
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	7.2823	0.875	8.322	0.000	5.567	8.998
budget	1.1122	0.017	67.117	0.000	1.080	1.145

Omnibus:	3349.953	**Durbin-Watson:**	1.321
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	69300.215
Skew:	2.727	**Prob(JB):**	0.00
Kurtosis:	19.993	**Cond. No.**	67.1

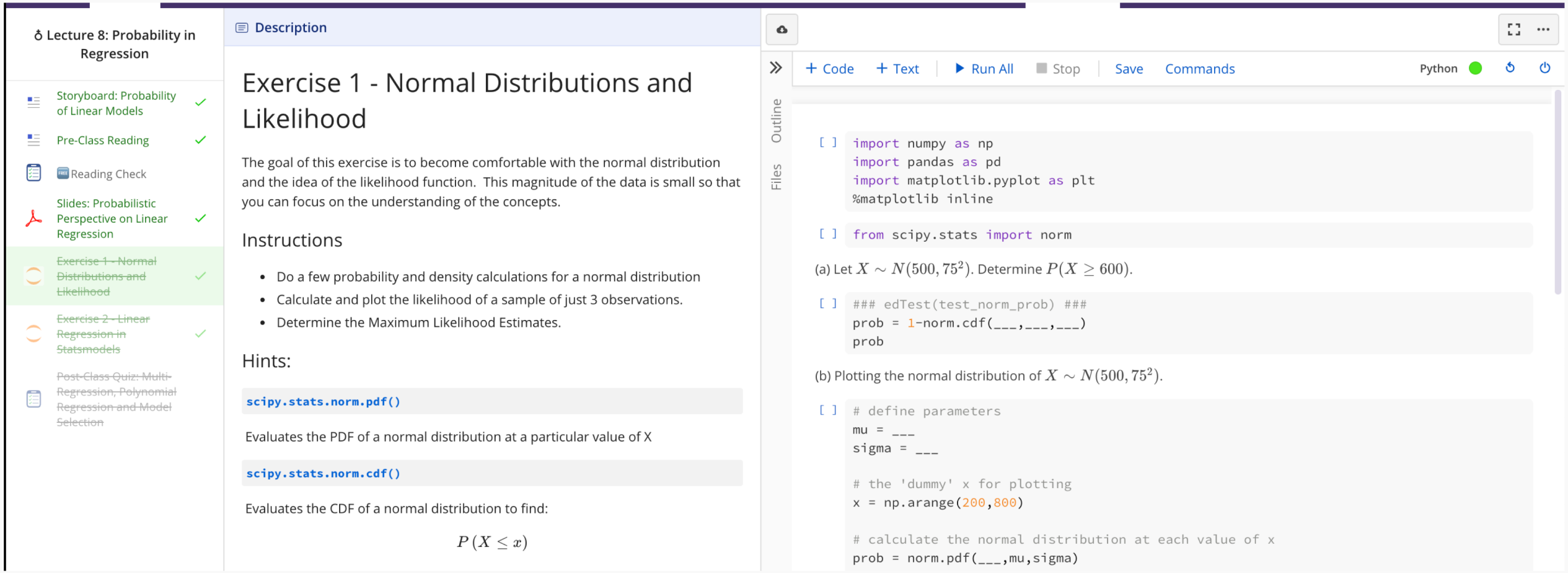


Exercise Time!

Ex. 1: Normal Distributions and Likelihoods (15 min)

Ex. 2: Linear Regression in `statsmodels` (15+ min)

Ex. 1: Normal Distributions and Likelihoods



Ex. 2: Linear Regression in `statsmodels`

Lecture 8: Probability in Regression

- Storyboard: Probability of Linear Models ✓
- Pre-Class Reading ✓
- Reading Check
- Slides: Probabilistic Perspective on Linear Regression ✓
- ~~Exercise 1 - Normal Distributions and Likelihood~~ ✓
- ~~Exercise 2 - Linear Regression in Statsmodels~~ ✓
- ~~Post-Class Quiz: Multi-Regression, Polynomial Regression and Model Selection~~

Description

Exercise 2 - Linear Regression in Statsmodels

The goal of this exercise is to use `statsmodels` to fit and interpret a regression model.

Roadmap

- Read the dataset 'movies.csv' as a dataframe
- Look at the scatterplot to predict revenue from budget costs, and investigate the appropriateness of linear regression.
- Estimate the linear regression model using both `sklearn` and `statsmodels`, and compare the results.
- Compare the coefficients of the multiple regression models (with additive effects and interactive effects) with those of the simple linear regression model.
- Investigate the appropriateness of the linear regression assumptions that the probabilisitic model implies.
- Bonus Question: calculate the log-likelihood for a linear regression model!

Hints

```
statmodels.OLS()
```

Fir an ordinary least squares (OLS) regression model using `statsmodels`.

Outline Files

+ Code + Text ▶ Run All ■ Stop Save Commands Python

```
[ ] # import libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

```
[ ] # Read the data from 'movies.csv' to a dataframe
movies = pd.read_csv('movies.csv')
```

```
[ ] #Take a peak at the dataset
movies.head()
```

(a) Plot the scatterplot to predict 'domestic' revenue from the 'budget' cost of the movie.

```
[ ] # create the scatterplot to predict 'domestic'
# from 'budget'
plt.scatter(___,___,marker='.')
plt.xlabel('Budget (millions $)')
plt.ylabel('Domestic Gross (millions $)')
plt.title('Revenue vs. Budget for Major Domestic Movies since 1980')
plt.show()
```

Question: What stands out in the plots above?Does linear regression seems appropriate based on this scatterplot?

(b) Use sklearn to get linear regression estimates.