Review Session

Milan, Italy.

Regression

Response vs. Predictor Variables

$X = X_1, \dots, X_p$

$X_j = x_{1j}, \dots, x_{ij}, \dots, x_{nj}$

predictors

features

covariates

independent variable

$\boldsymbol{y} = y_1, \dots, y_i, \dots, y_n$

outcome

response variable

dependent variable

n observations

TV	radio	newspaper	sales
230.1	37.8	69.2	22.1
44.5	39.3	45.1	10.4
17.2	45.9	69.3	9.3
151.5	41.3	58.5	18.5
180.8	10.8	58.4	12.9

p predictors

Response vs. Predictor Variables

This is called X: a.k.a.
The Design Matrix

y:
The response variable

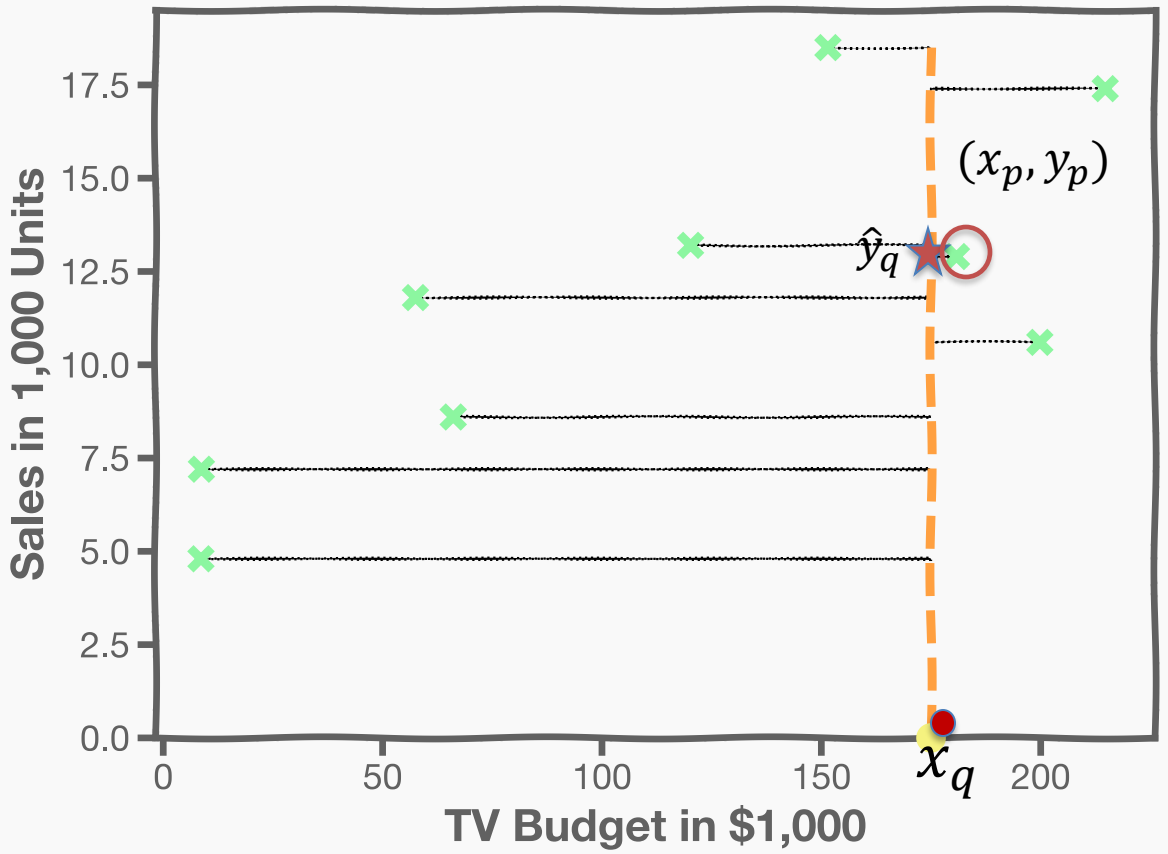
n observations

TV	radio	newspaper
230.1	37.8	69.2
44.5	39.3	45.1
17.2	45.9	69.3
151.5	41.3	58.5
180.8	10.8	58.4

sales
22.1
10.4
9.3
18.5
12.9

kNN

k-Nearest Neighbors – kNN



What is $\hat{y}_q$ at some x_q ?

Find distances to all other points $D(x_q, x_i)$

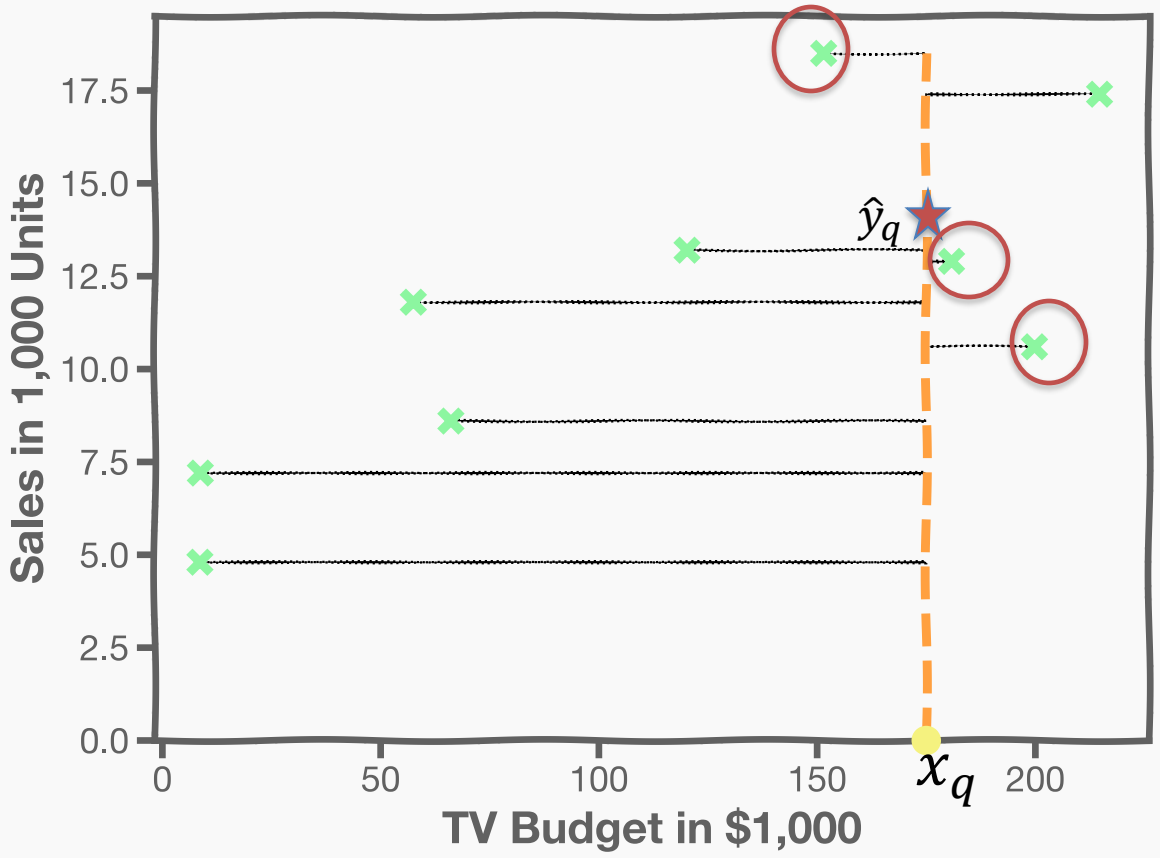
Find the nearest neighbor, (x_p, y_p)

Predict $\hat{y}_q = y_p$

k-Nearest Neighbors – kNN

Do the same for "all" $x's$

k-Nearest Neighbors – kNN



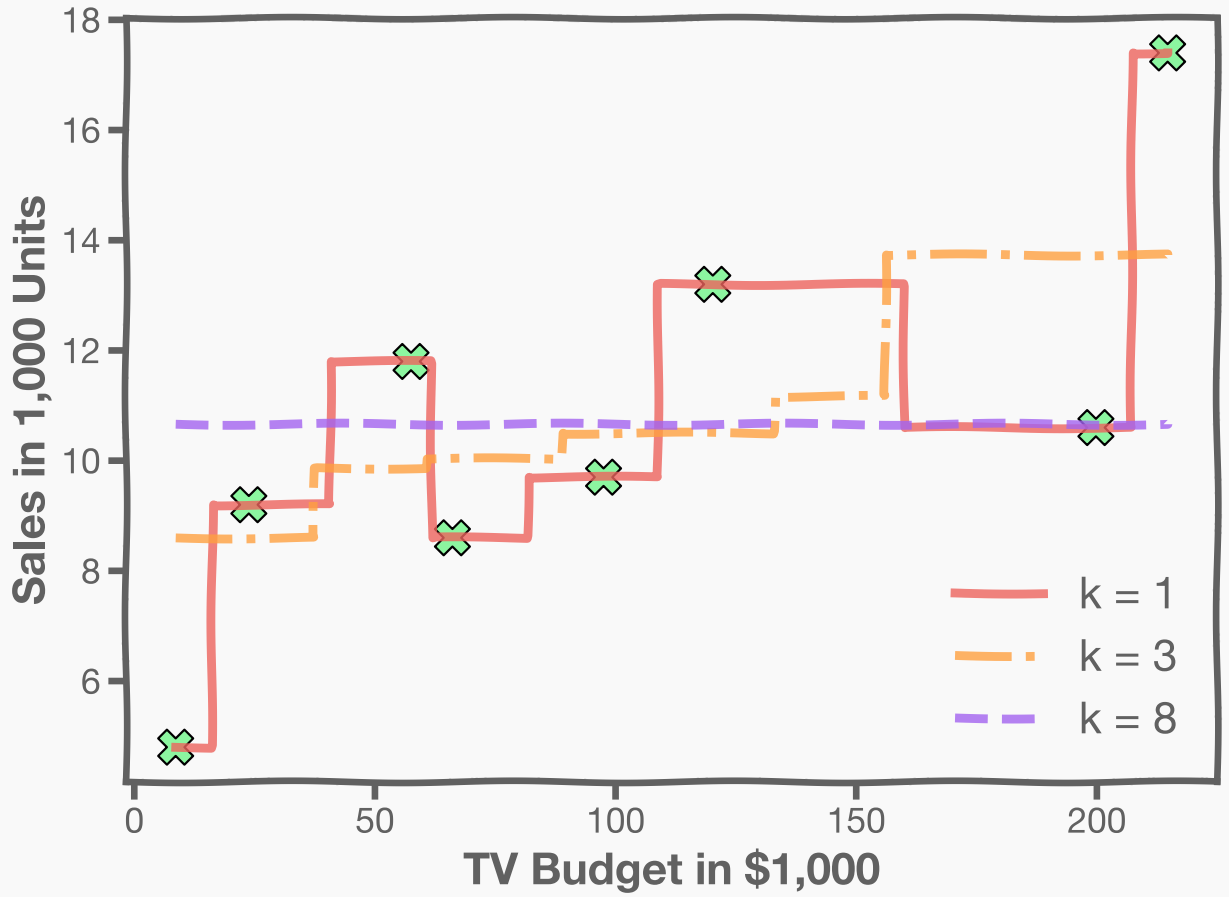
What is $\hat{y}_q$ at some x_q ?

Find distances to all other points $D(x_q, x_i)$

Find the k-nearest neighbors, $x_{q_1}, \ldots, x_{q_k}$

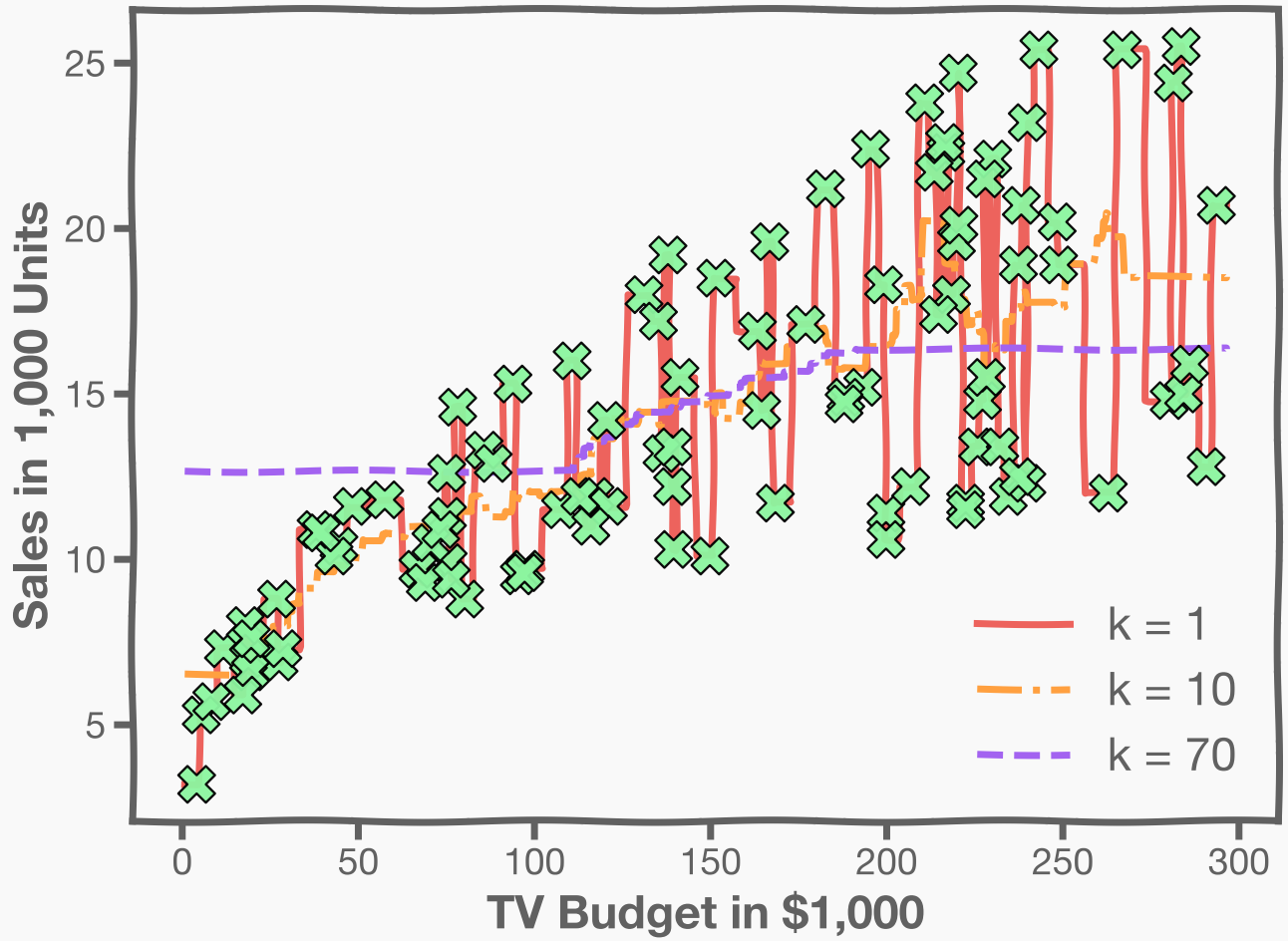
Predict $\hat{y}_q = \frac{1}{k}\sum_i^k y_{q_i}$

k-Nearest Neighbors – kNN



k-Nearest Neighbors – kNN

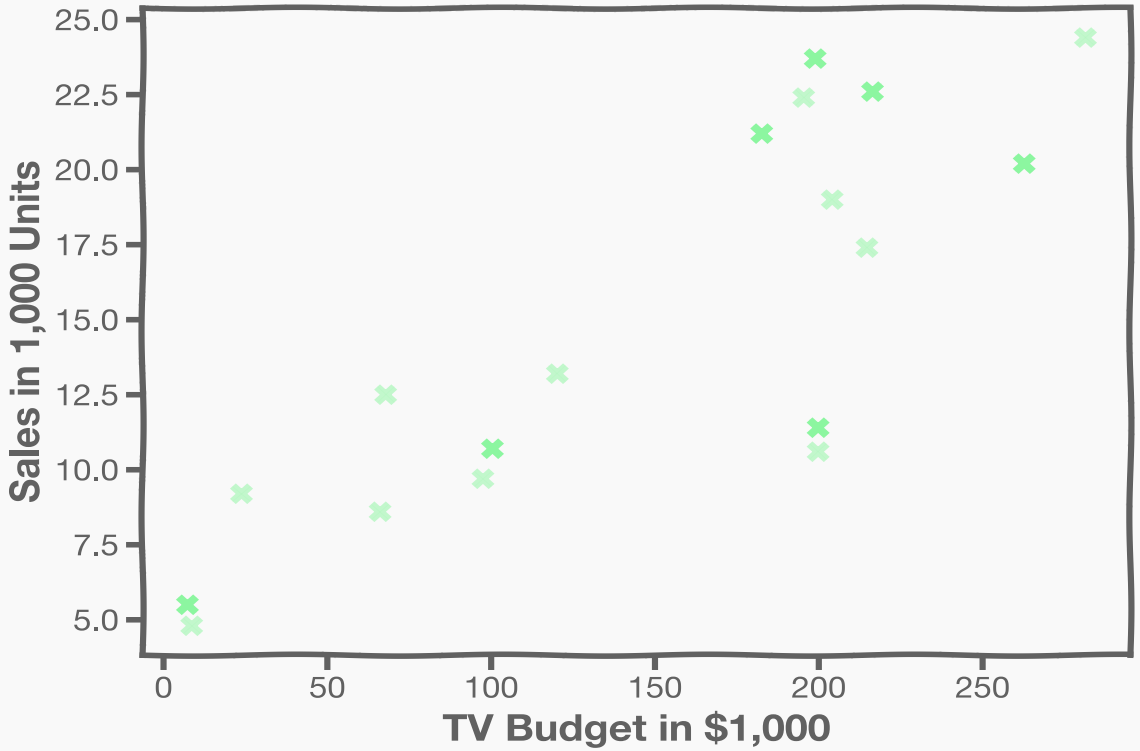
We can try different k-models on more data



Error Evaluation

Error Evaluation

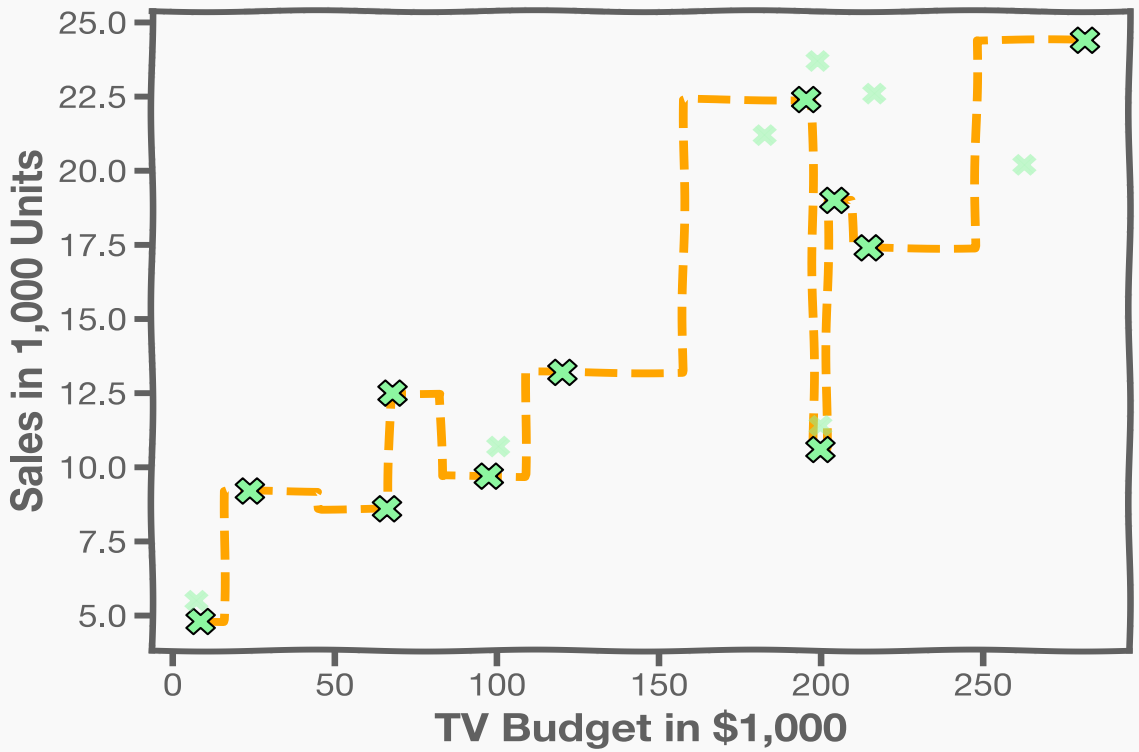
We first **withhold** a portion of the data from the model; this process is called **train-test** split.



We use the training set to estimate $\hat{y}$, and the test set to evaluate the model's performance.

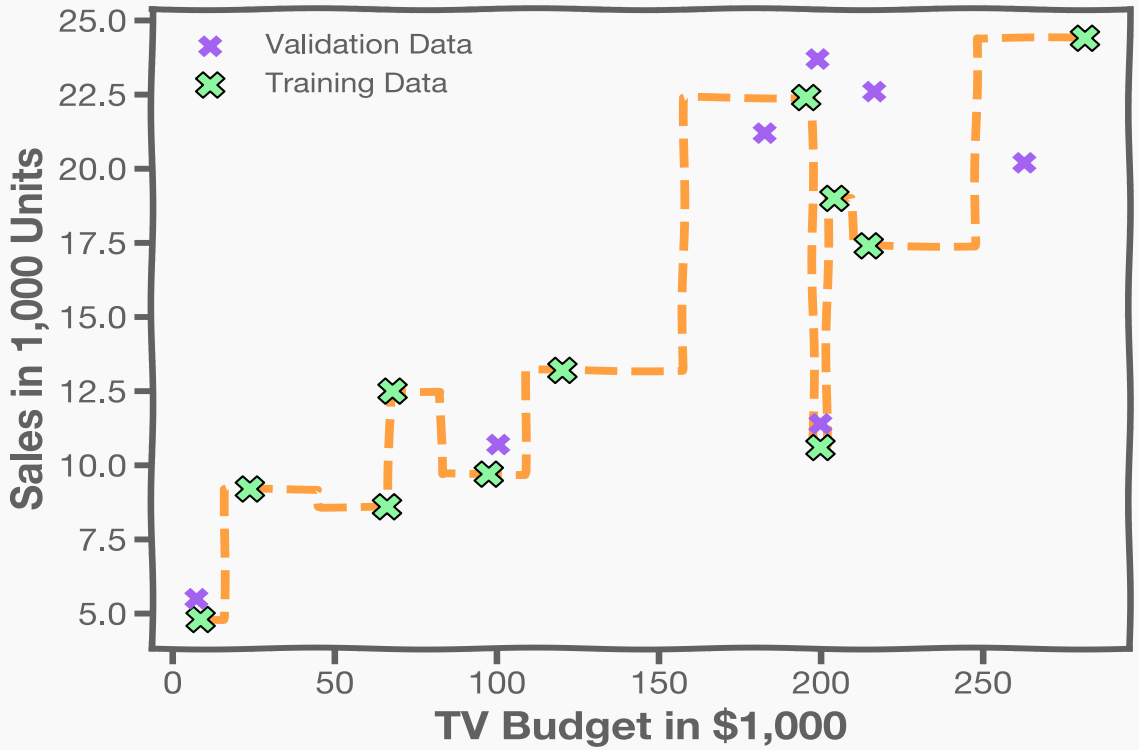
Error Evaluation

Estimate $\hat{y}'s$ values for all the data points in the training set when ***k=1.***



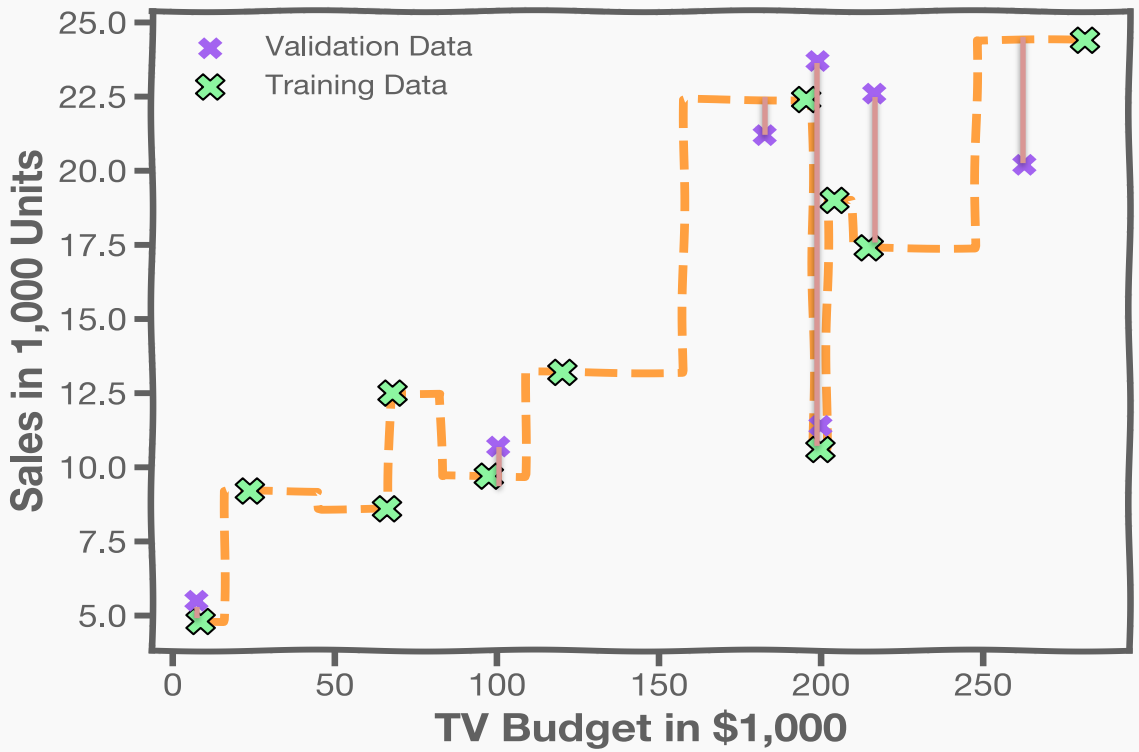
Error Evaluation

Now, we examine the data that was not used for estimating $\hat{y}$, the **test data** represented by purple crosses.



Error Evaluation

And we calculate the **residuals** $(y_i - \hat{y}_i)$.



For each observation (x_n, y_n), the absolute residuals, $r_i = |y_i - \hat{y}_i|$ quantify the error at each observation point.

Error Evaluation

To quantify the performance of a model, we aggregate the errors. This aggregated value is commonly referred to as the ***loss***, ***error***, or ***cost function***.

A widely used loss function for quantitative outcomes is the **Mean Squared Error (MSE):**

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Note: Loss and cost function refer to the same thing. Cost usually refers to the total loss where loss refers to a single training point.

R-squared

$$R^2 = 1 - \frac{\sum_i (\hat{y}_i - y_i)^2}{\sum_i (\bar{y} - y_i)^2}$$

- If our model is as good as the mean value, $\bar{y}$, then $R^2 = 0$
- If our model is perfect, then $R^2 = 1$
- R^2 can be negative if the model is worse than the average. This can happen when we evaluate the model in the **test** set.

Simple Linear Regression

True vs. Statistical Model

We assume that the response variable, Y, is related to the predictor variables, X, through an unknown function which can be generally expressed as:

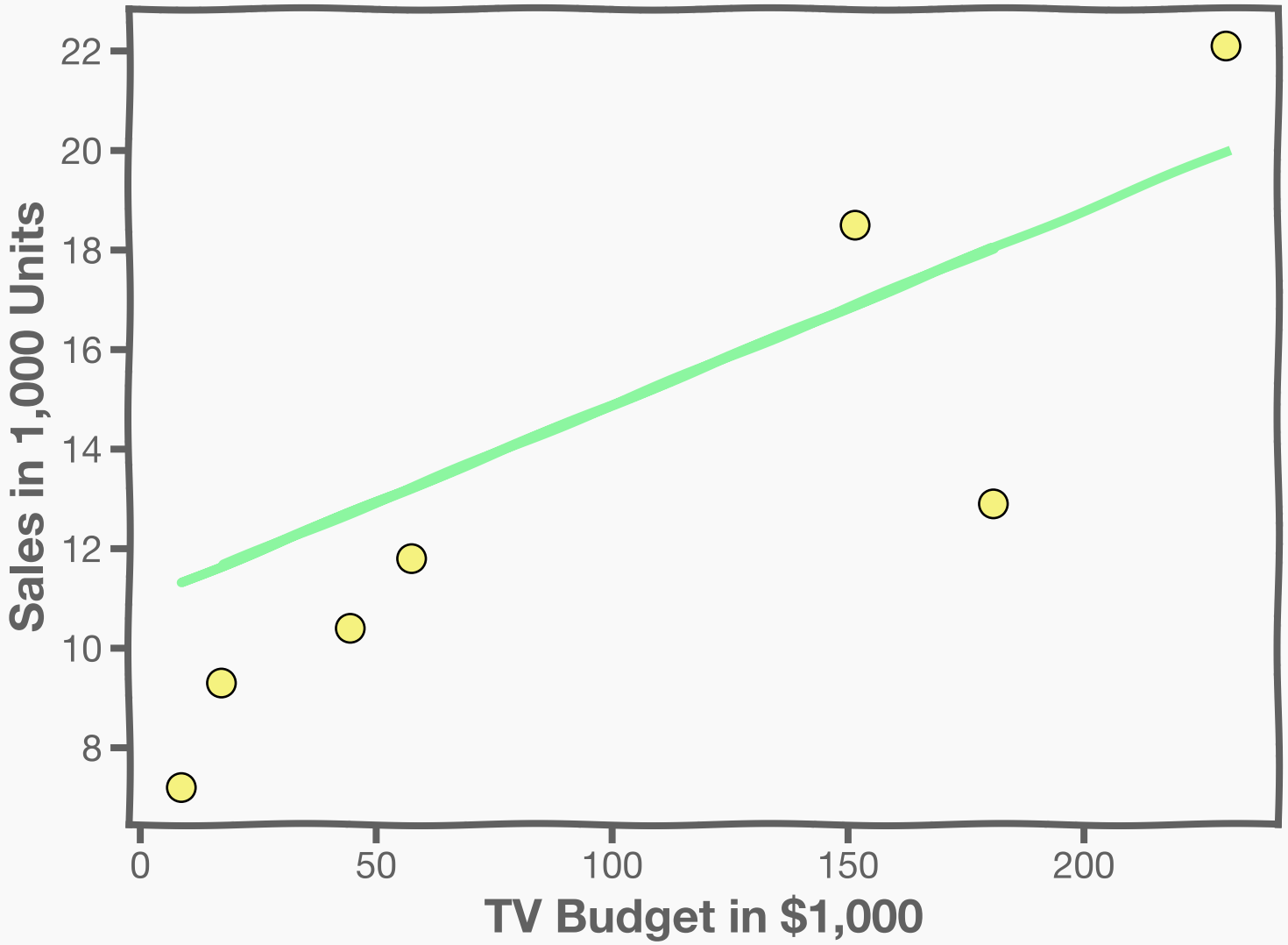
$$Y = f(X) + \varepsilon$$

Here, f represents the unknown function expressing an underlying rule for relating Y to X. ε represents the random amount (unrelated to X) that Y differs from the rule $f(X)$.

A ***statistical model*** is any algorithm used to estimate f. We denote the estimated function as $\hat{f}$.

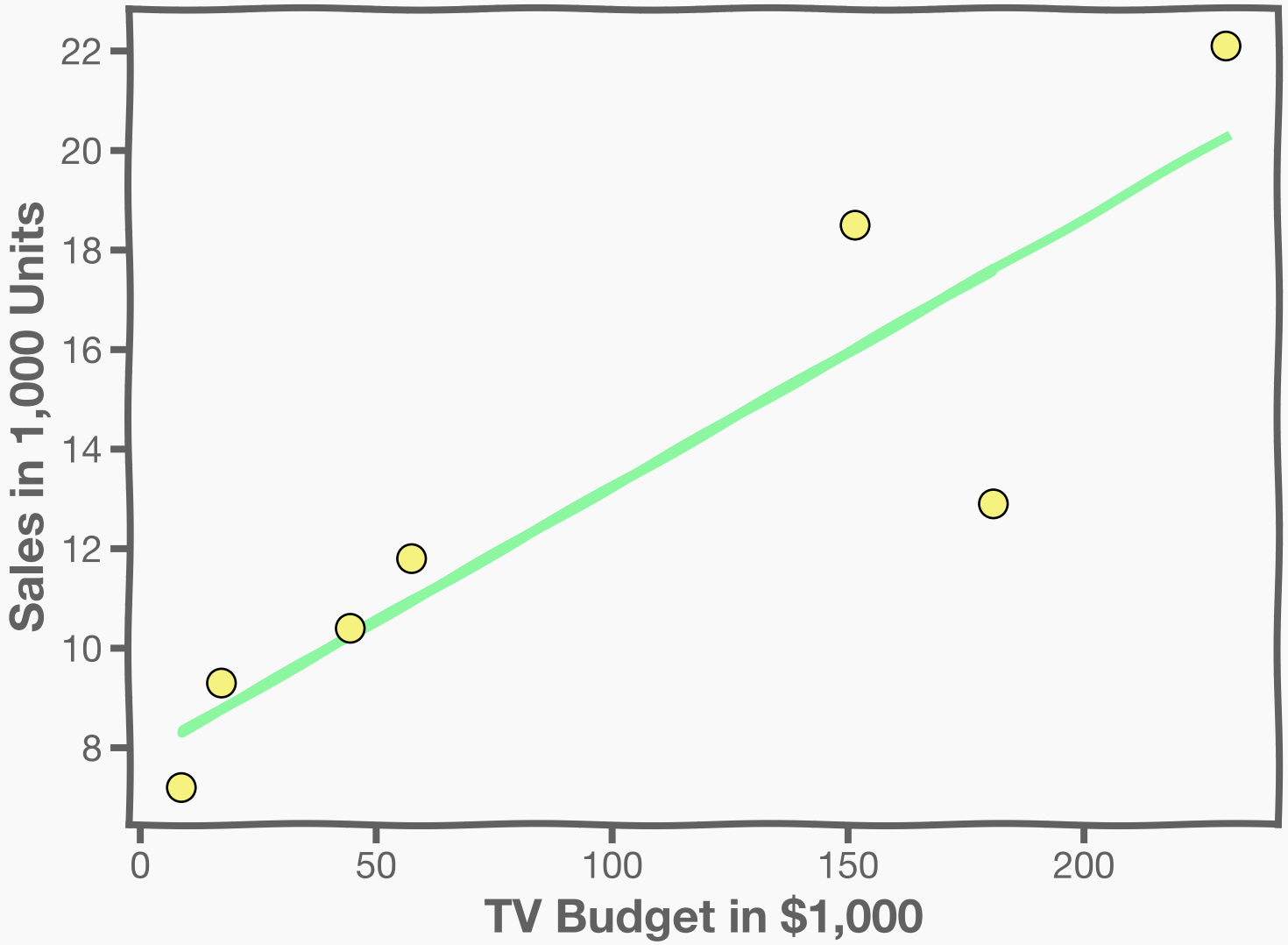
Estimate of the regression coefficients (cont)

Is this line good?



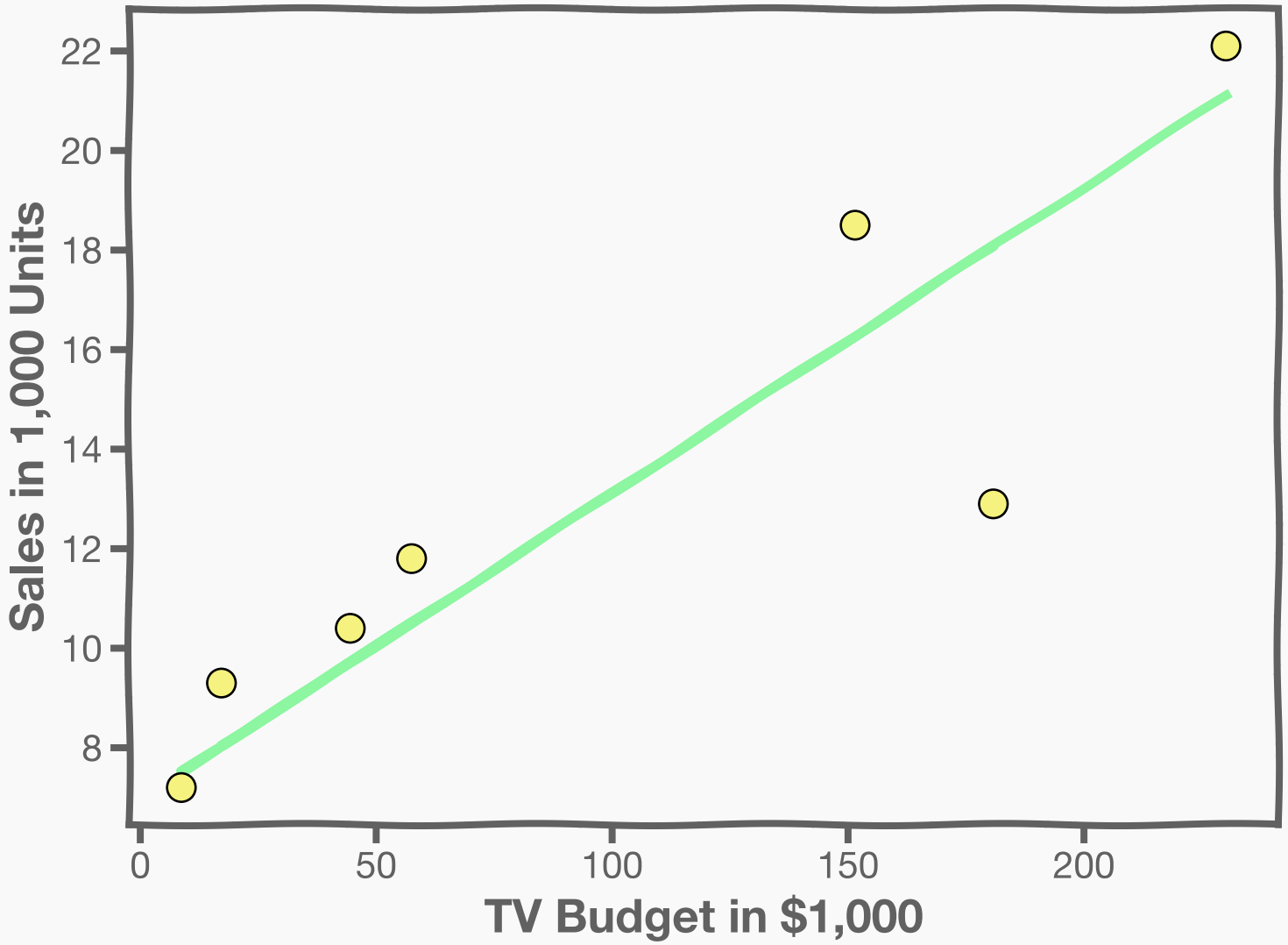
Estimate of the regression coefficients (cont)

Maybe this one?



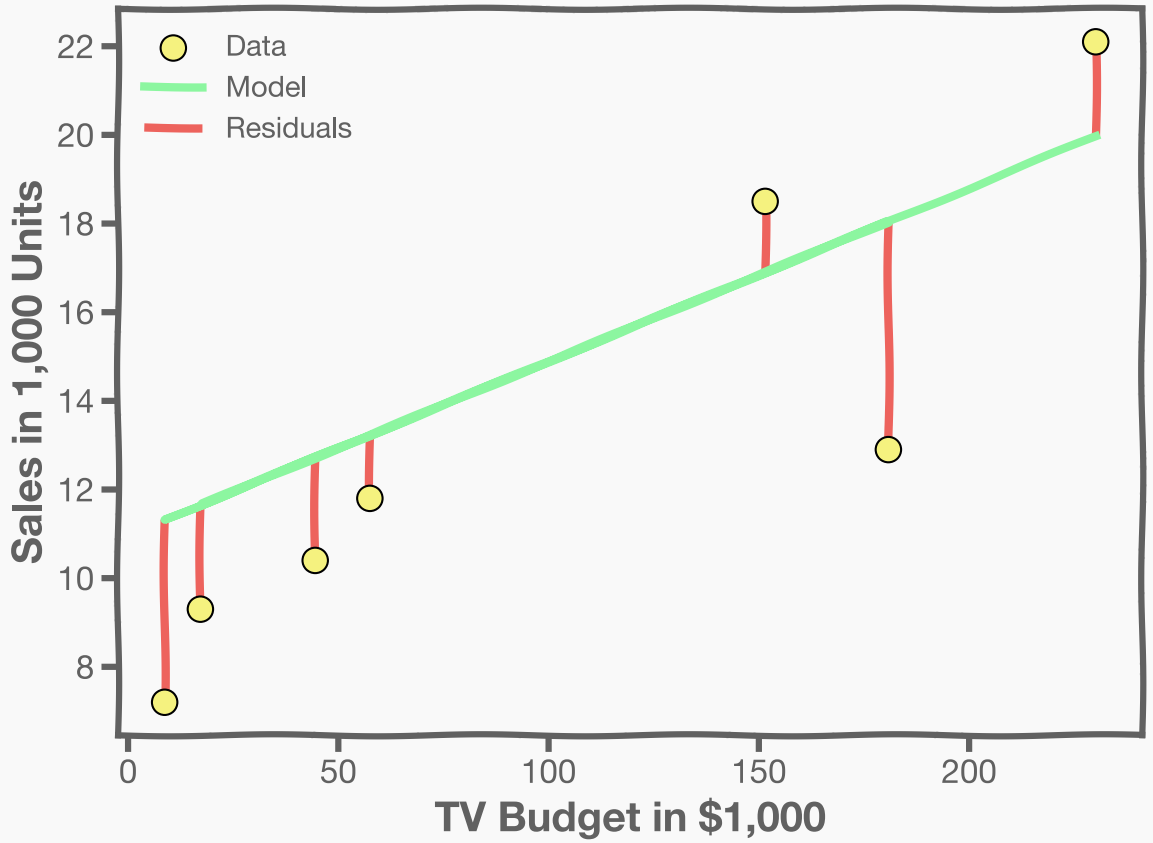
Estimate of the regression coefficients (cont)

Or this one?



Estimate of the regression coefficients (cont.)

Question: Which line is the best?



As before, for each observation (x_n, y_n), the absolute residuals, $r_i = |y_i - \hat{y}_i|$ quantify the error at each observation.

Summary: Estimate of the regression coefficients

We use MSE as our **loss function**,

$$L\left(\beta_0, \beta_1\right) = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \widehat{y}_i\right)^2 = \frac{1}{n}\sum_{i=1}^{n}\left[y_i - \left(\beta_1 x_i + \beta_0\right)\right]^2$$

We choose $\hat{\beta}_1$ and $\hat{\beta}_0$ in order to minimize the predictive errors made by our model, i.e. minimize our loss function.

Then the optimal values for $\hat{\beta}_0$ and $\hat{\beta}_1$ should be:

FIND THE VALUES OF β_0 AND β_1 THAT YIELD THE SMALLEST VALUE OF L

$$\widehat{\beta}_0, \widehat{\beta}_1 = \underset{\beta_0, \beta_1}{\operatorname{argmin}} L(\beta_0, \beta_1).$$

WE CALL THIS **FITTING** OR **TRAINING** THE MODEL

Estimate of the regression coefficients: analytical solution

Take the gradient of the loss function and find the values of $\hat{\beta}_0$ and $\hat{\beta}_1$ where the gradient is zero: $\nabla L = \left[\frac{\partial L}{\partial \beta_0}, \frac{\partial L}{\partial \beta_1}\right]$=0

$$\hat{\beta}_1 = \frac{\sum_i (x_i - \overline{x})(y_i - \overline{y})}{\sum_i (x_i - \overline{x})^2}$$

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

where $\bar{y}$ and $\bar{x}$ are sample means.

The line:

is called the **regression line**.

$$\widehat{Y} = \widehat{\beta}_1 X + \widehat{\beta}_0$$

Assumptions of Linear Regression

Linearity: Relationship between variables is linear.

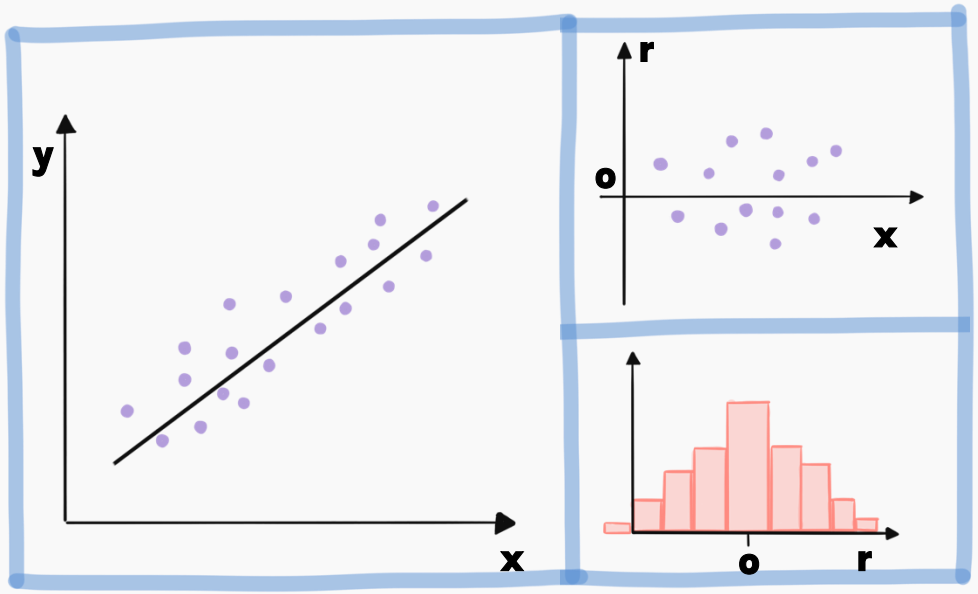
$$f(x) = \beta_0 + \beta_1 x$$

Independence: No correlation between errors.
Homoscedasticity: Constant variance of residuals.
Normality of Residuals: Residuals are normally distributed.

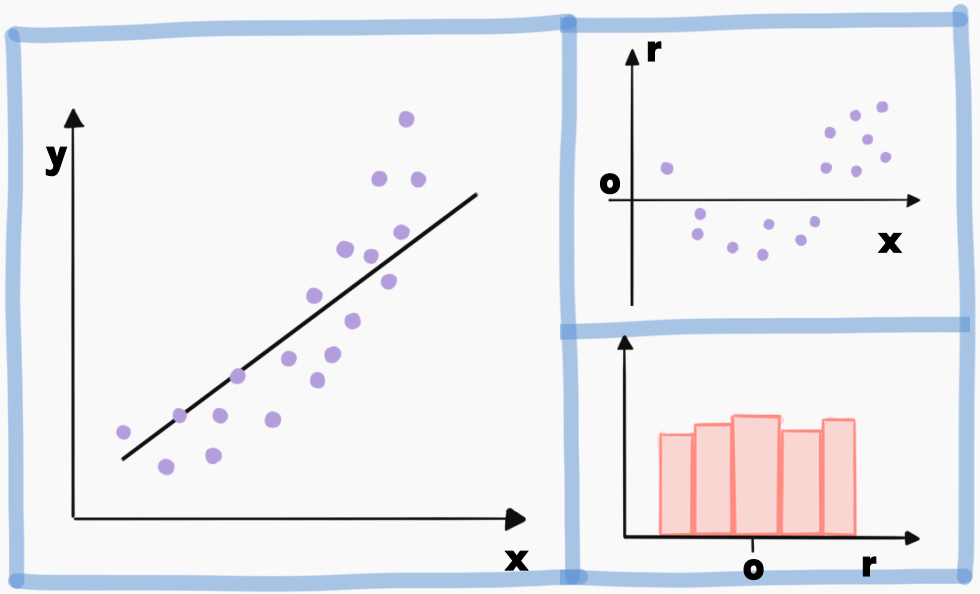
$$y = f(x) + \epsilon$$
$$L(\beta_0, \beta_1) = MSE$$

Other things to consider
Fixed X: Independent variables are error-free.
No Multicollinearity: Low correlation between predictors.

Residual Analysis



Linear relationship plausible. There is no obvious relationship between residuals and x. Histogram of residuals is symmetric and normally distributed.



Linear relationship unlikely. There is an obvious relationship between residuals and x. Histogram of residuals is symmetric but **not** normally distributed.

Note: For multi-regression, we plot the residuals vs predicted, $\hat{y}$, since there are too many x's and that could wash out the relationship.

Multiple Linear Regression

RECAP: Multi-Linear Regression

The model takes a simple algebraic form: $\mathbf{Y} = \mathbf{X}\beta + \epsilon$

We will again choose the **MSE** as our loss function, which can be expressed in vector notation as

$$\mathrm{MSE}(\beta) = \frac{1}{\mathrm{n}} \|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|^2$$

Minimizing the MSE using vector calculus yields,

$$\widehat{\boldsymbol{\beta}} = \left(\mathbf{X}^\top \mathbf{X}\right)^{-1} \mathbf{X}^\top \mathbf{Y} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}}\, \mathrm{MSE}(\boldsymbol{\beta}).$$

Understanding Scaling: Standardization & Normalization

Scaling transforms your data so that it fits within a specific range or distribution.

Standardization (Z-Score)

Transforms data to have mean = 0 and standard deviation = 1

$$\frac{X - mean}{std}$$

Normalization (Min-Max Scaling)

Rescales data to range between 0 and 1

$$\frac{X - min}{max - min}$$

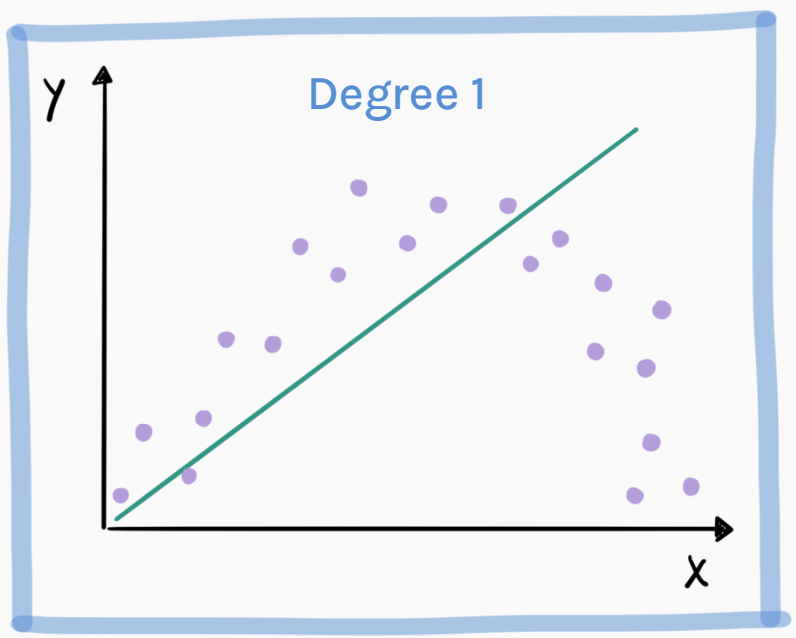
Why Scale?

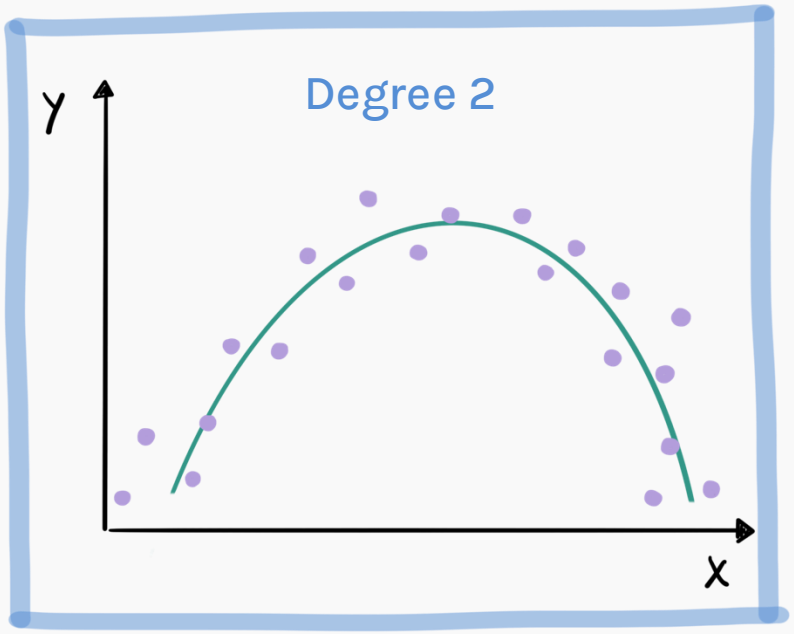
- Makes algorithms sensitive to feature scales perform better
- Facilitates easier interpretation and analysis

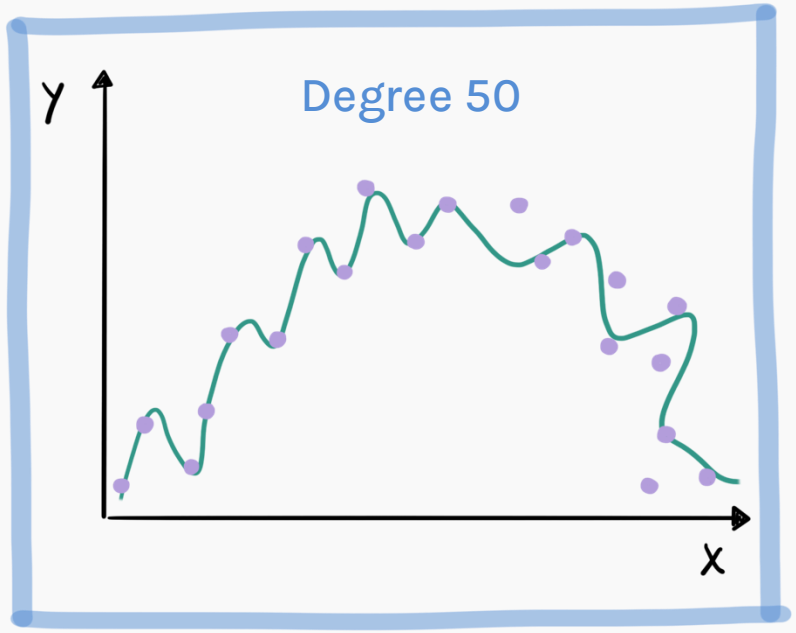
📚 **For More In-depth c**heck my notes and examples on EdStem!

Polynomial Regression (cont.)

Fitting a polynomial model requires choosing a degree.







Underfitting: when the degree is too low, the model cannot fit the trend.

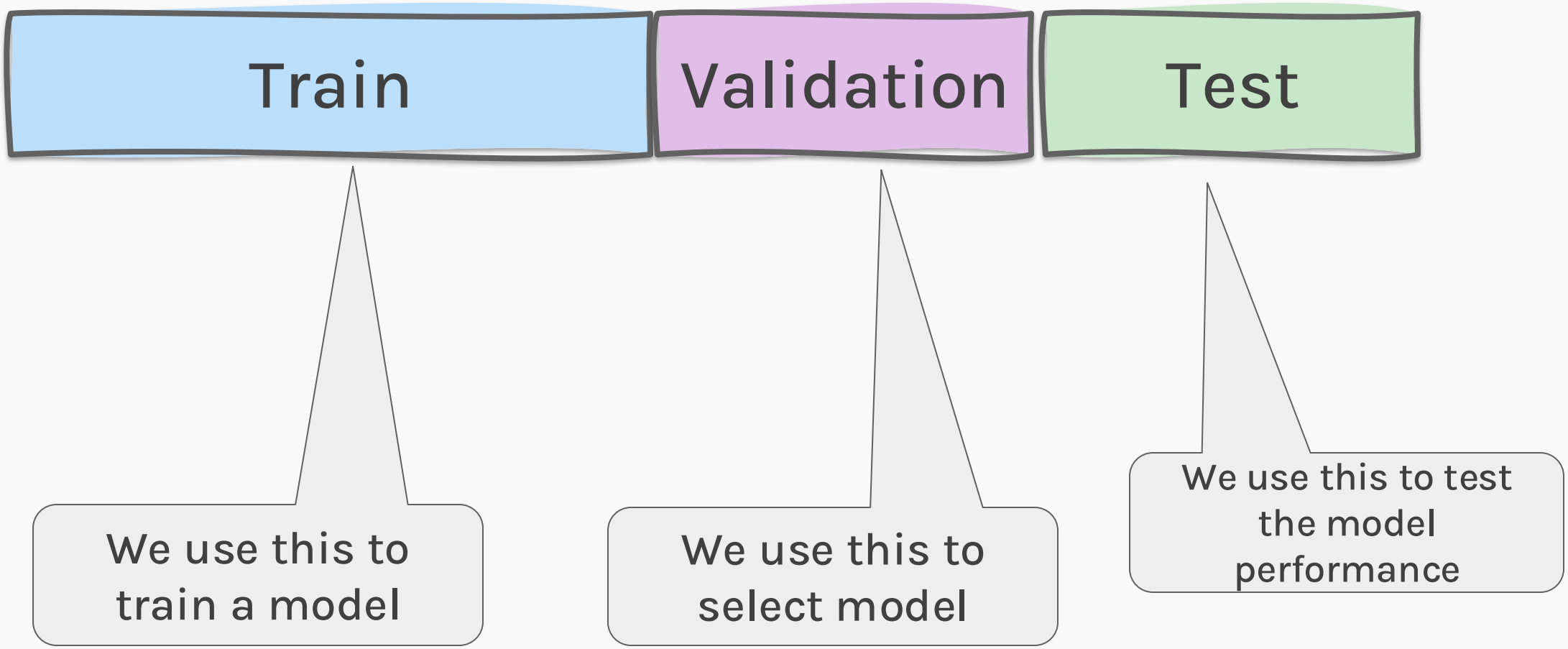
We want a model that fits the trend and ignores the noise.

Overfitting: when the degree is too high, the model fits all the noisy data points.

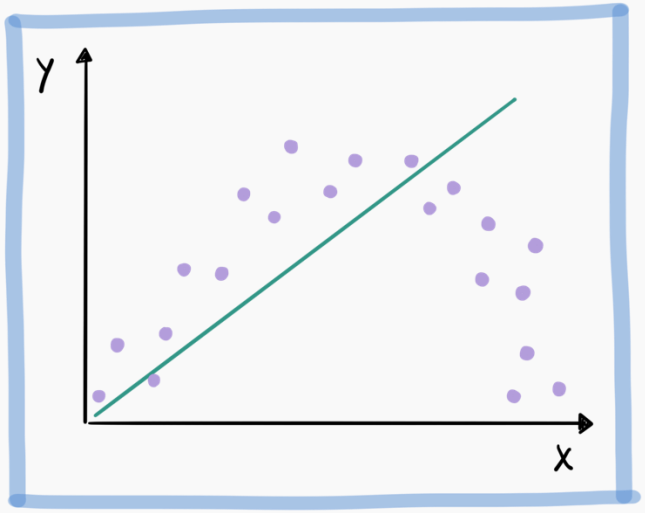
Cross-Validation

Train-Validation-Test

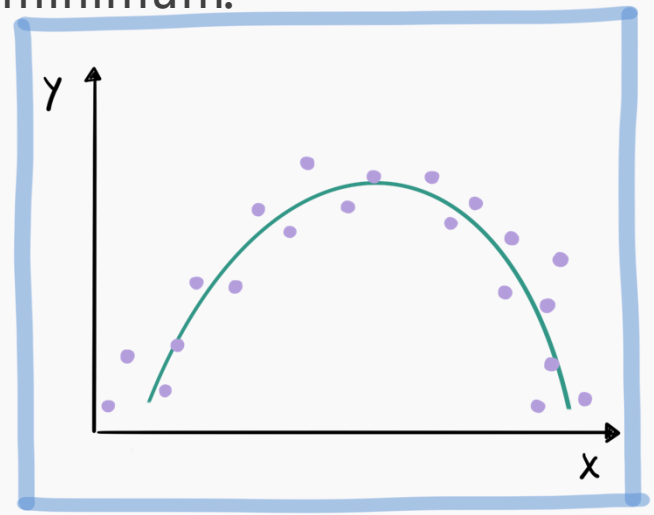
We introduced a different sub-set, which we called validation and we use it to select the model.

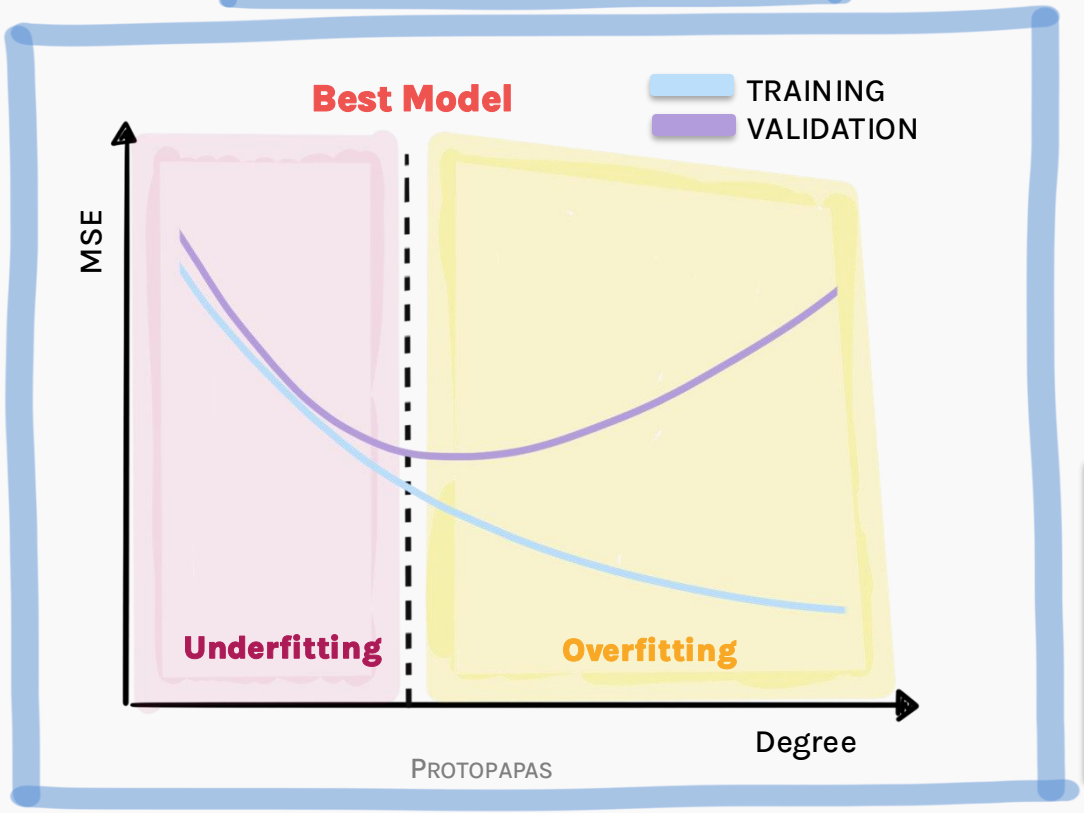


Underfitting: train and validation error is high.

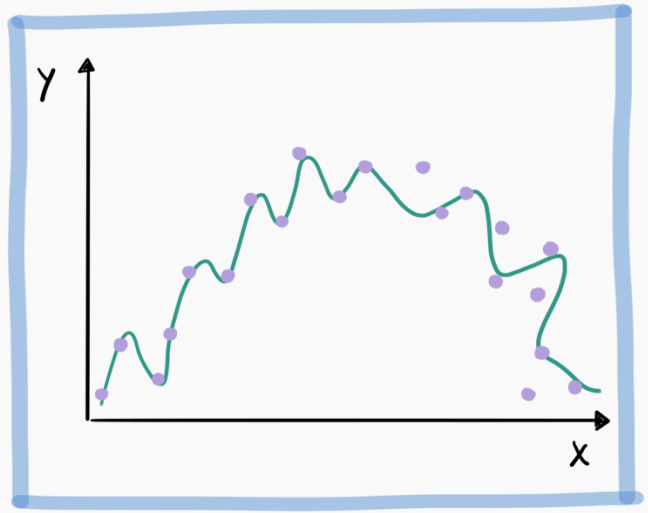


Best model: validation error is minimum.



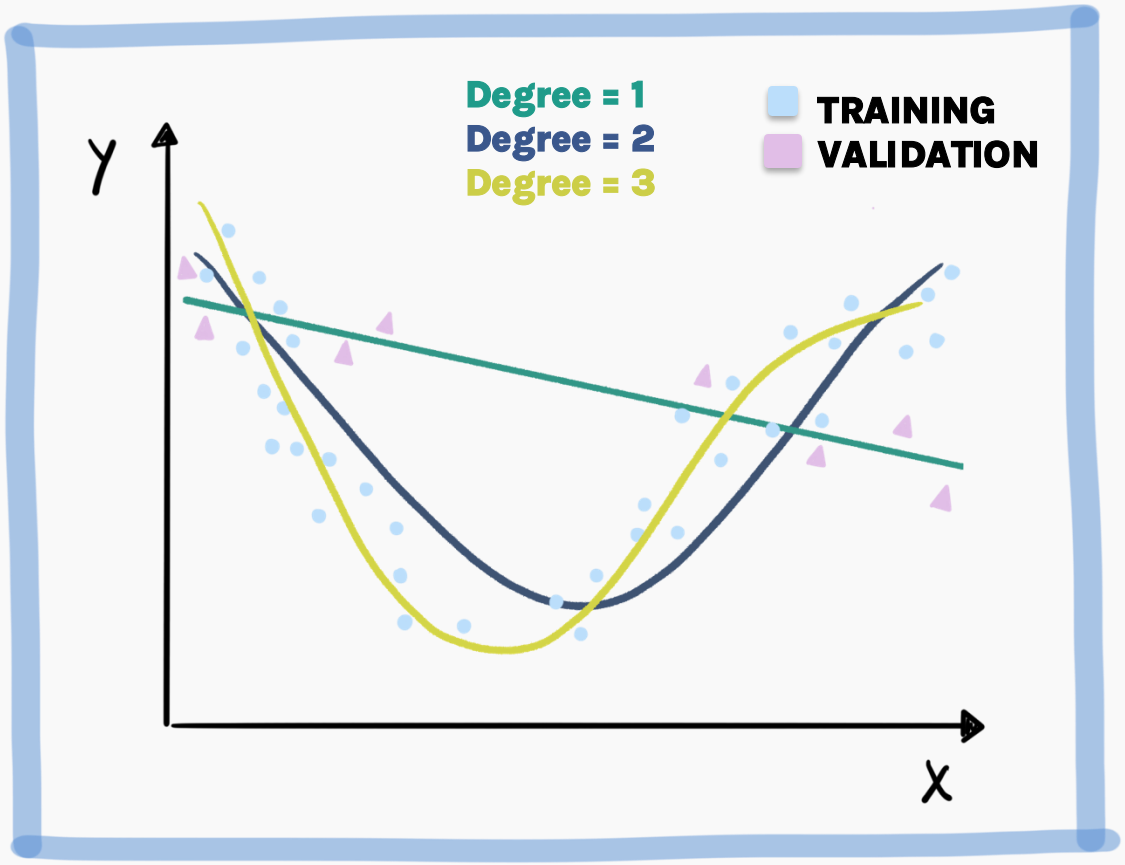


Overfitting: train error is low, validation error is high.



What are the parameters of the models and what are the hyperparameters?

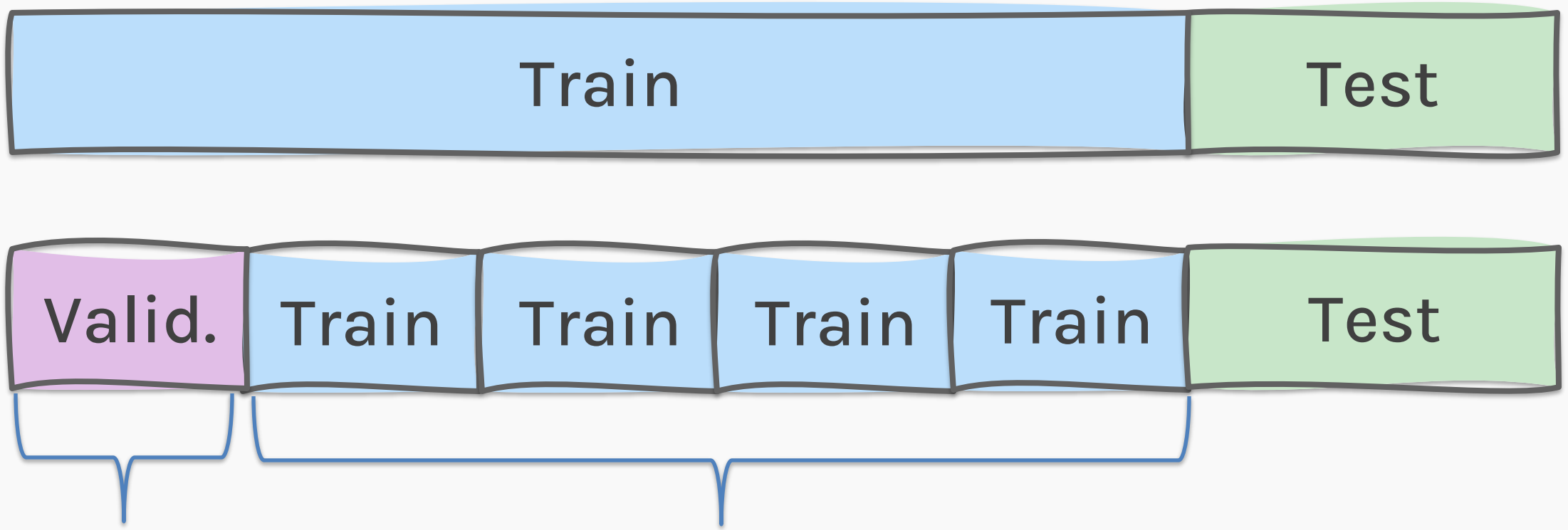
Cross Validation: Motivation



It is obvious that degree=3 is the correct model, but the validation set by chance favors the linear model.

Using a single validation set to select amongst multiple models can be problematic - **there is the possibility of overfitting to the validation set.**

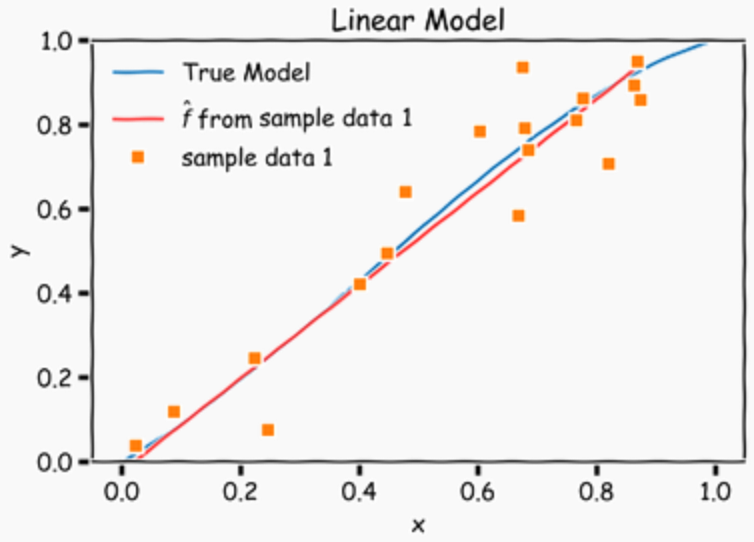
Cross Validation

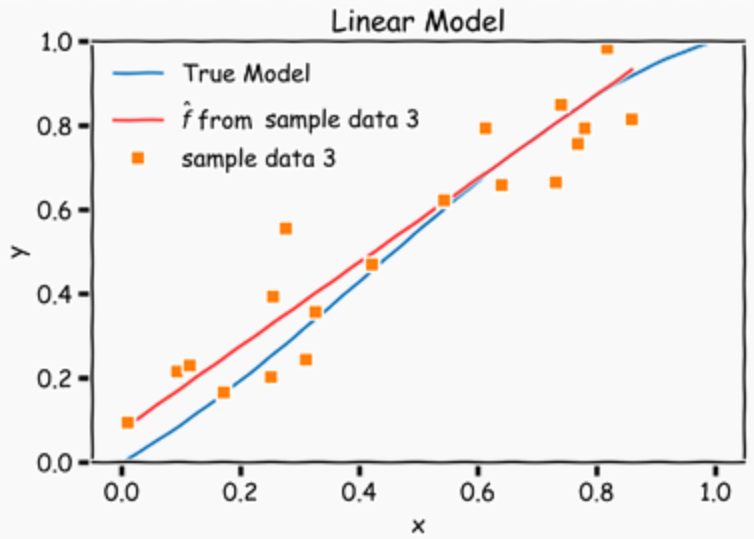


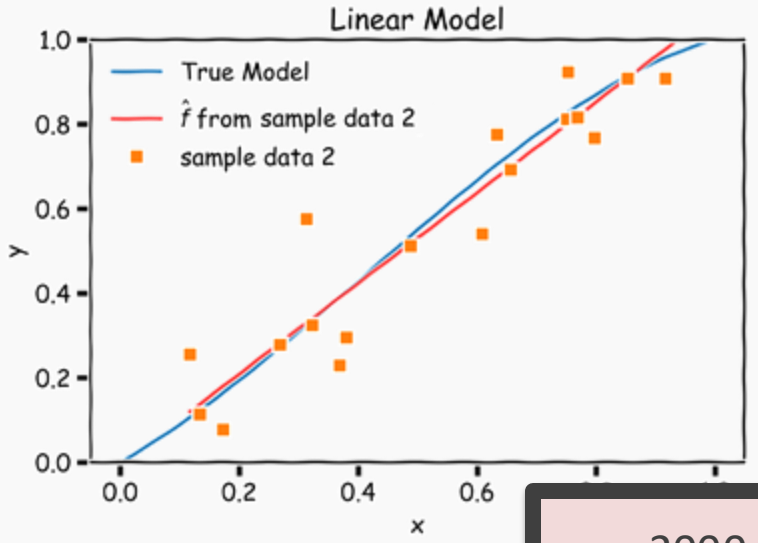
$$MSE^{val} = \frac{1}{5}\sum_{i=1}^{5} MSE_i^{val}$$

Bias-Variance Trade-Off

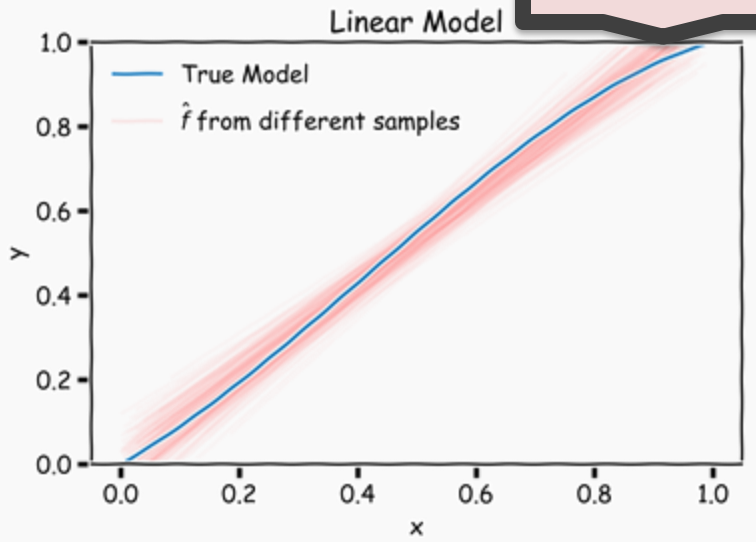
Bias vs Variance: Variance of a SIMPLE model



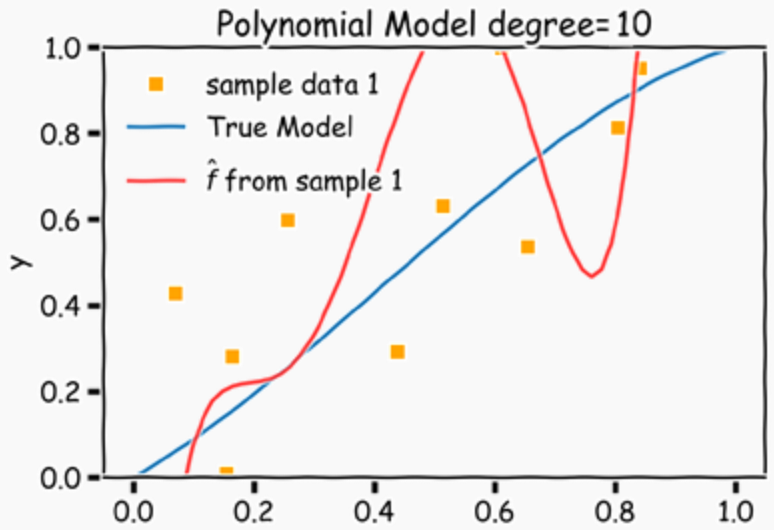


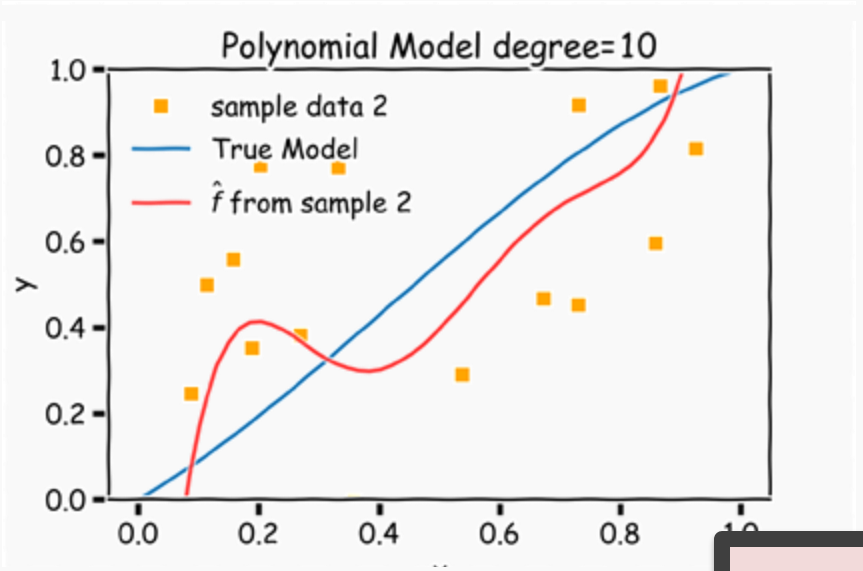


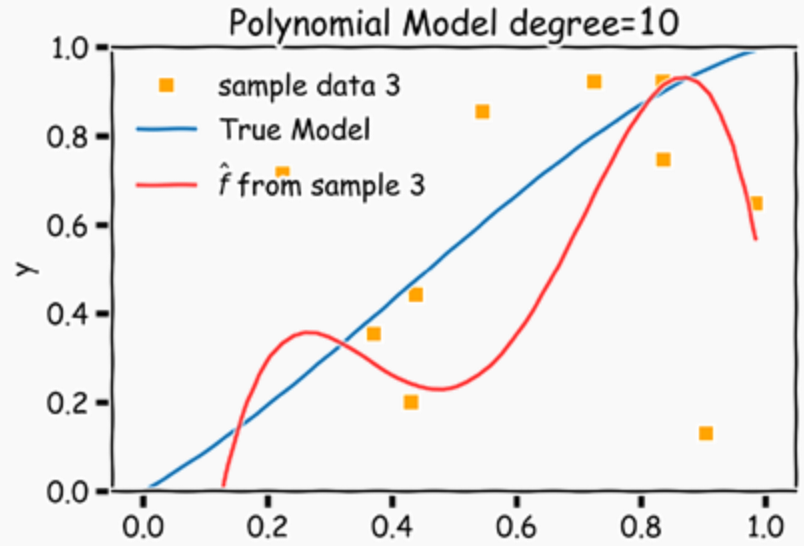
2000 models

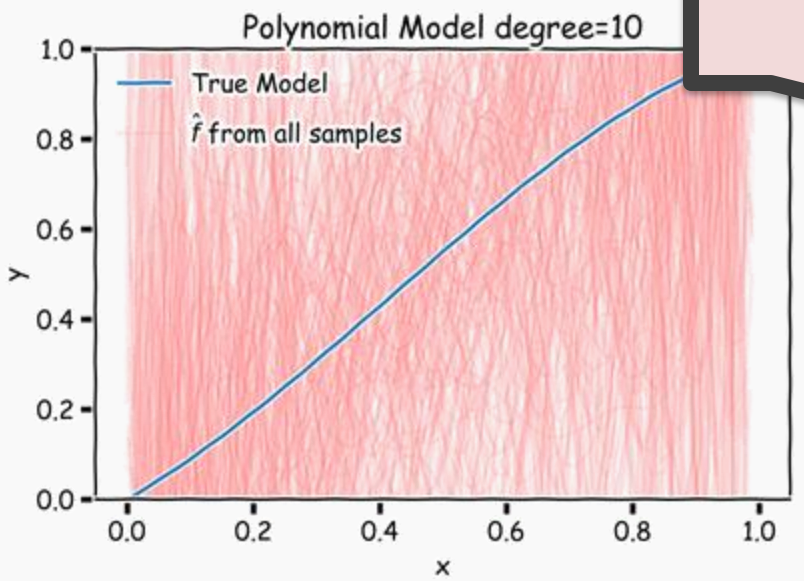


Bias vs Variance: Variance of a COMPLEX model





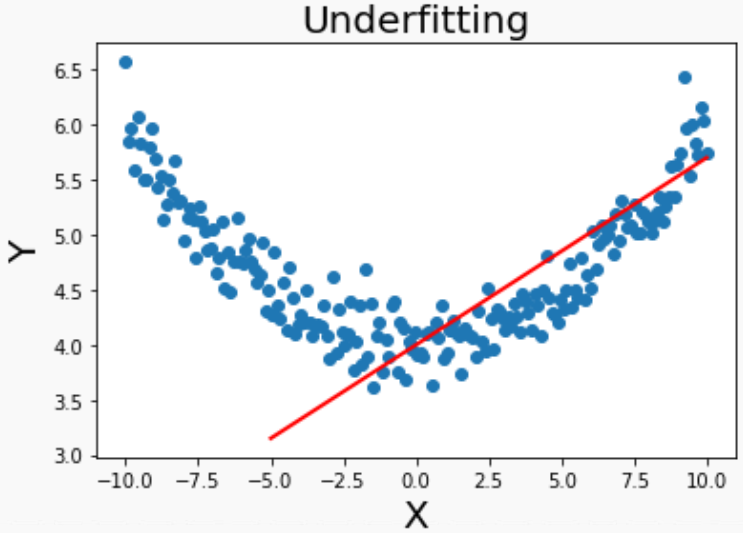


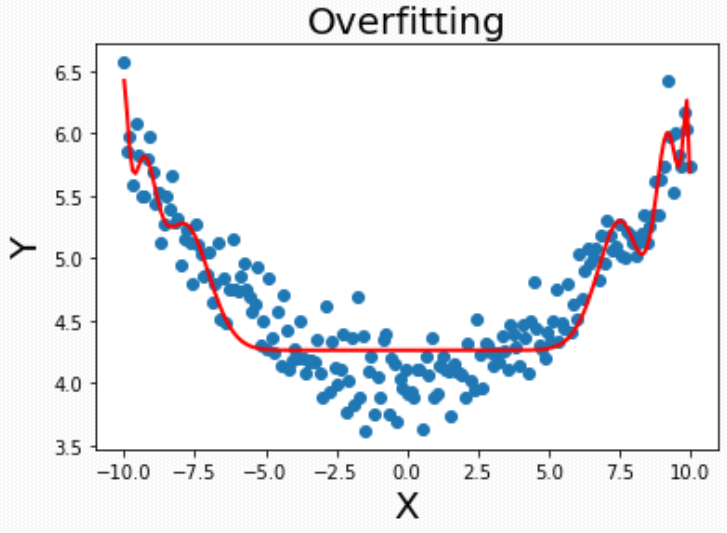


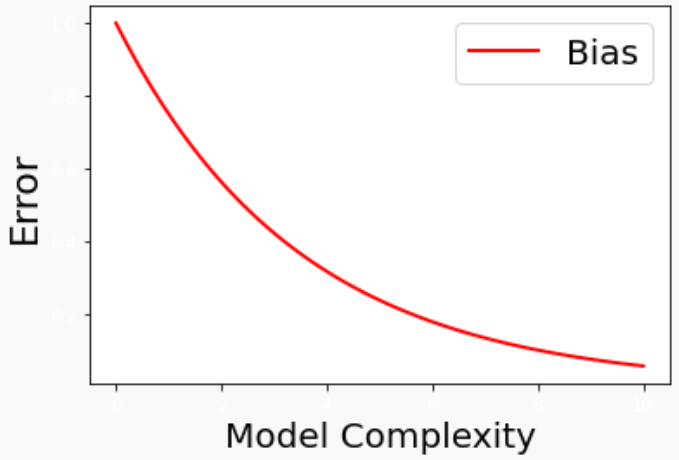
2000 models

The Bias-Variance Trade Off

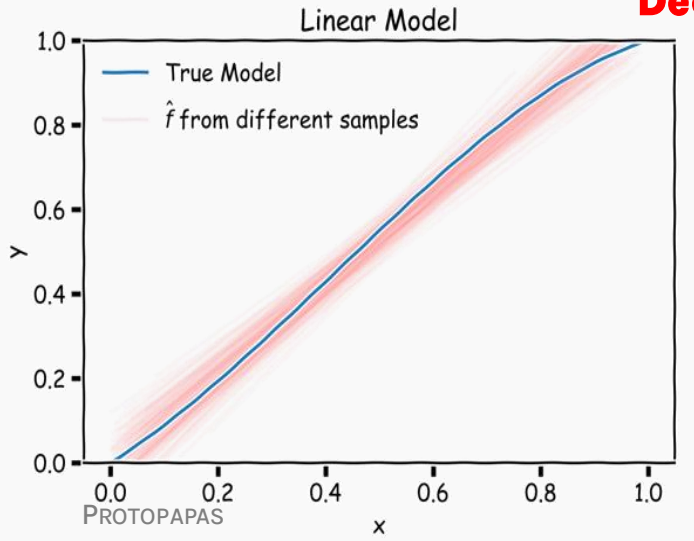
Increase complexity

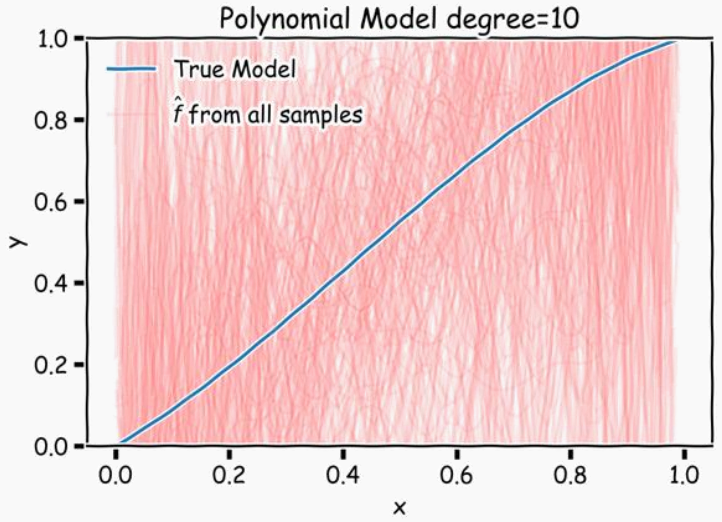


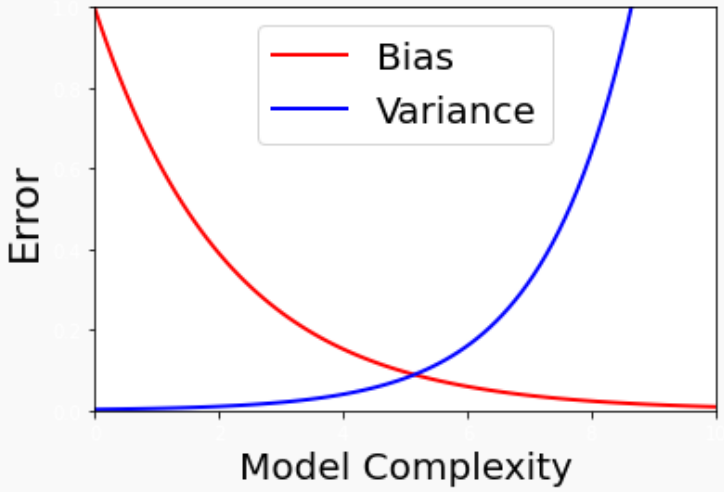




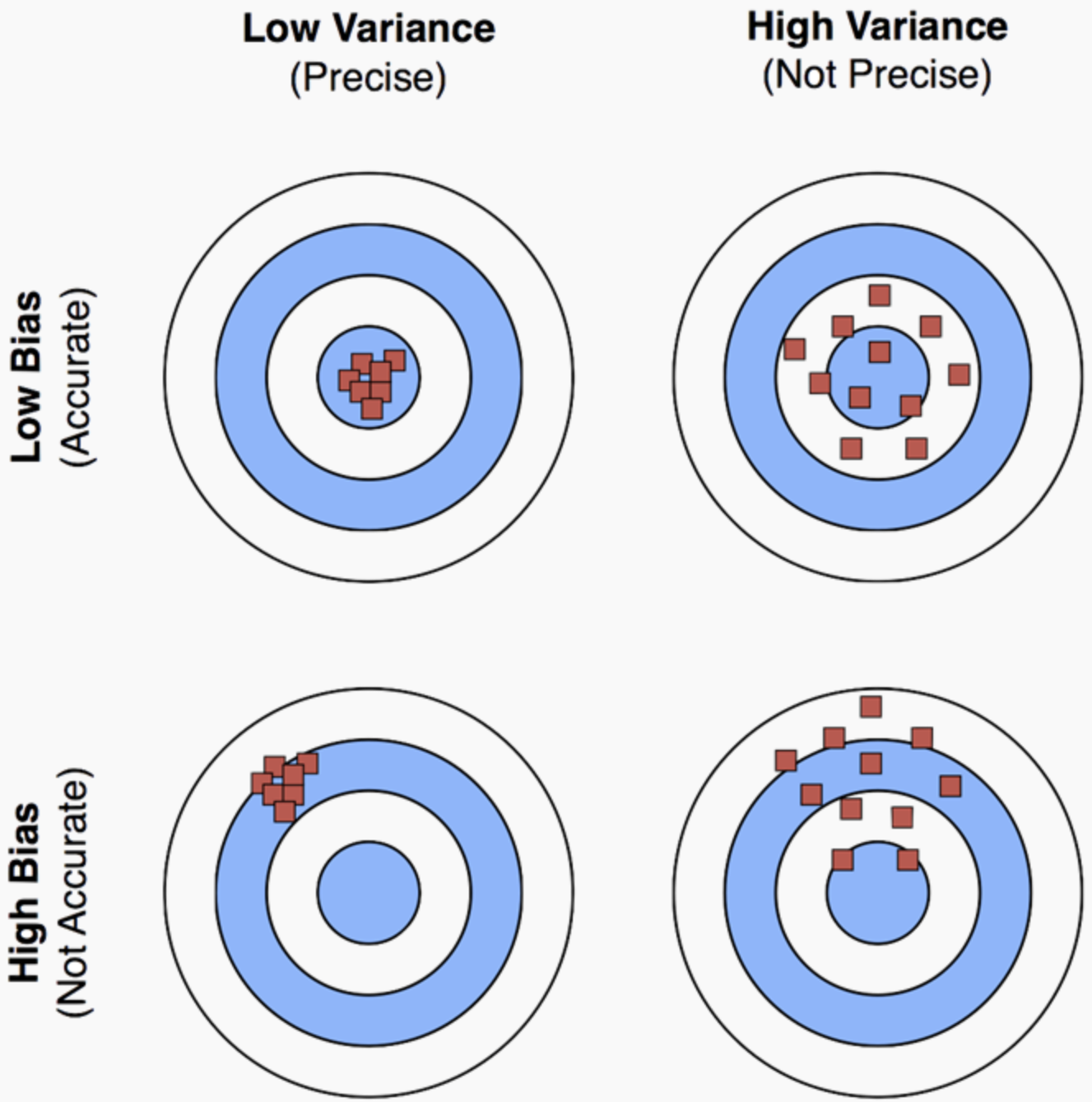
Decrease complexity







WE WANT THIS



WE WANT TO AVOID THIS

Regularization

Regularization

What we want

Low model error

Minimize:

$$\frac{1}{n}\sum_{i=1}^{n}\left|y_i - \boldsymbol{\beta}^\top \boldsymbol{x}_i\right|^2$$

Discourage extreme values in model parameters

Minimize:

$$L_{reg} = \begin{cases} \sum_{j=1}^{J} \beta_j^2 \\ \sum_{j=1}^{J} |\beta_j| \end{cases}$$

How do we combine these two objectives?

Regularization: Ridge Regression

Ridge regression: minimize $\mathcal{L}_{RIDGE}$ with respect to $\beta's$

$$\mathcal{L}_{RIDGE} = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \boldsymbol{\beta}^\top \boldsymbol{x}_i\right|^2 + \lambda \sum_{j=1}^{J} \beta_j^2$$

No need to regularize the bias, β_0, since it is not connected to the predictors.

Regularization: LASSO Regression

Lasso regression: minimize $\mathcal{L}_{LASSO}$ with respect to $\beta's$

$$\mathcal{L}_{LASSO} = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \boldsymbol{\beta}^\top \boldsymbol{x}_i\right|^2 + \boldsymbol{\lambda}\sum_{j=1}^{J}|\beta_j|$$

Regularization

What we want

Low model error

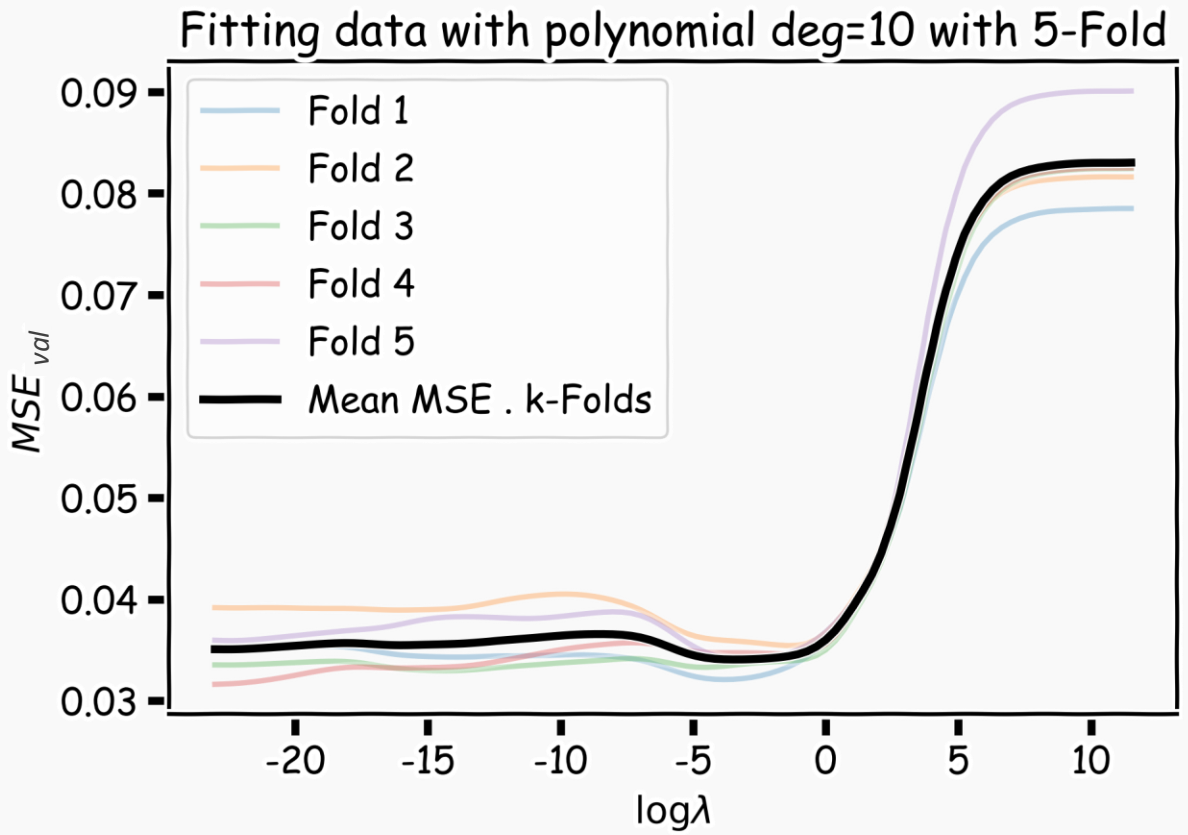
Discourage extreme values in model parameters

Minimize: ze:

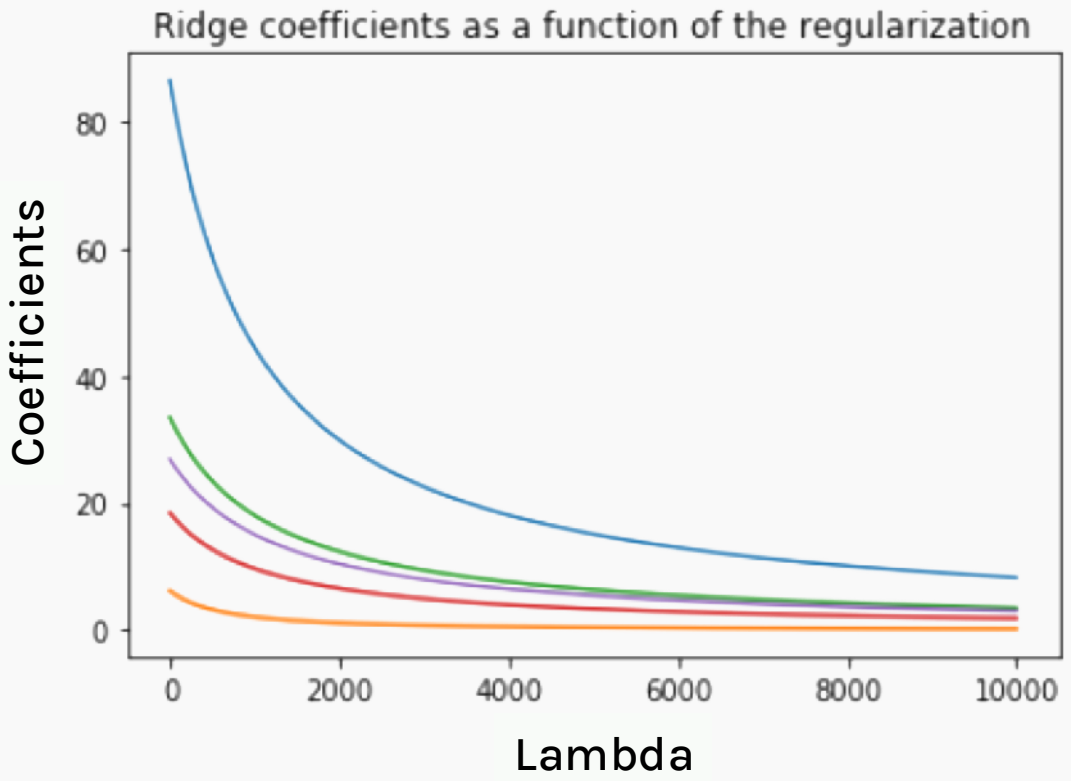
λ is the **regularization parameter,** balancing model error and the penalty term.
We select λ with cross-validation

$$\mathcal{L}_{REG} = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \boldsymbol{\beta}^\top \boldsymbol{x}_i\right|^2 + \lambda\ L_{reg}$$

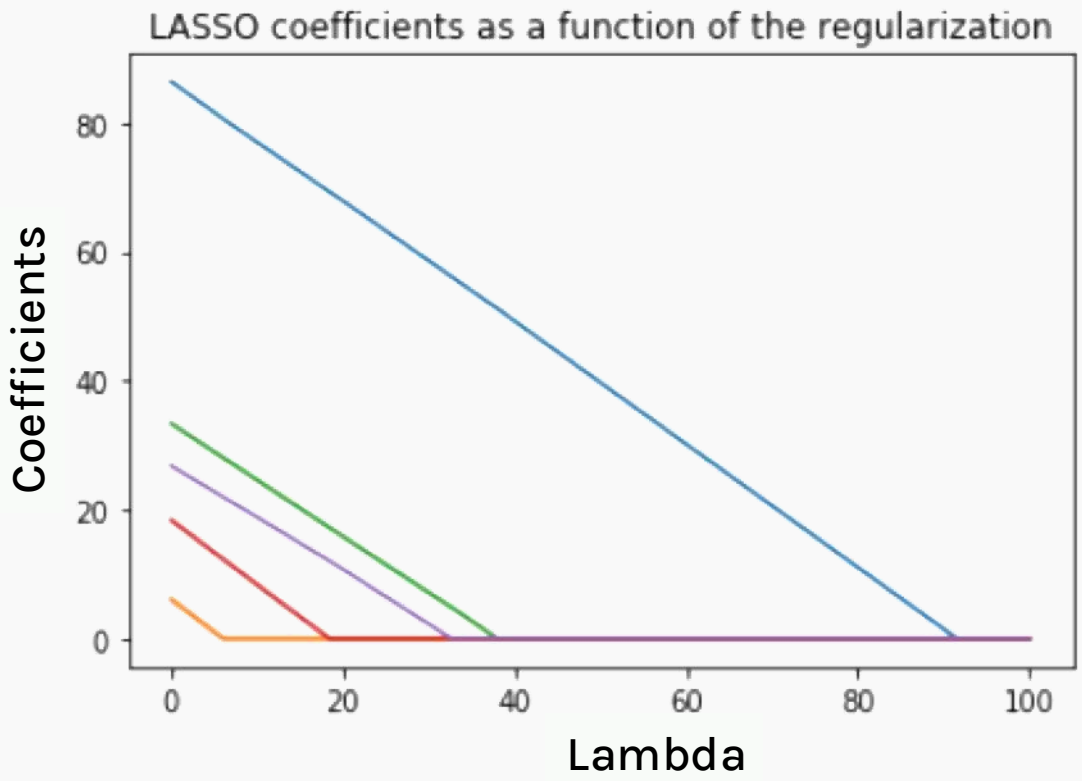
Ridge regularization with **cross-validation** only: step by step



Ridge and LASSO visualized



The values of the coefficients decrease as lambda increases, but they are not nullified.



The values of the coefficients decrease as lambda increases and are nullified fast.

Probabilistic Interpretation of Linear Regression:

MSE from Maximum Likelihood

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:

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

For any random variable, ϵ, that has zero mean, then:

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

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

The Probabilistic Regression Model

If we assume that $\epsilon_i \sim N(0, \sigma^2)$

This regression model can be rewritten as:

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

The likelihood of a measurement having value Y_i given X_i for a model β_0, β_1:

$$L(\beta_0, \beta_1, \sigma^2|Y_i, X_i) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}\left(\frac{Y_i - (\beta_0 + \beta_1 X_i)}{\sigma}\right)^2}$$

The Probabilistic Regression Model

The likelihood of a measurement having value Y_i given X_i for a model β_0, β_1

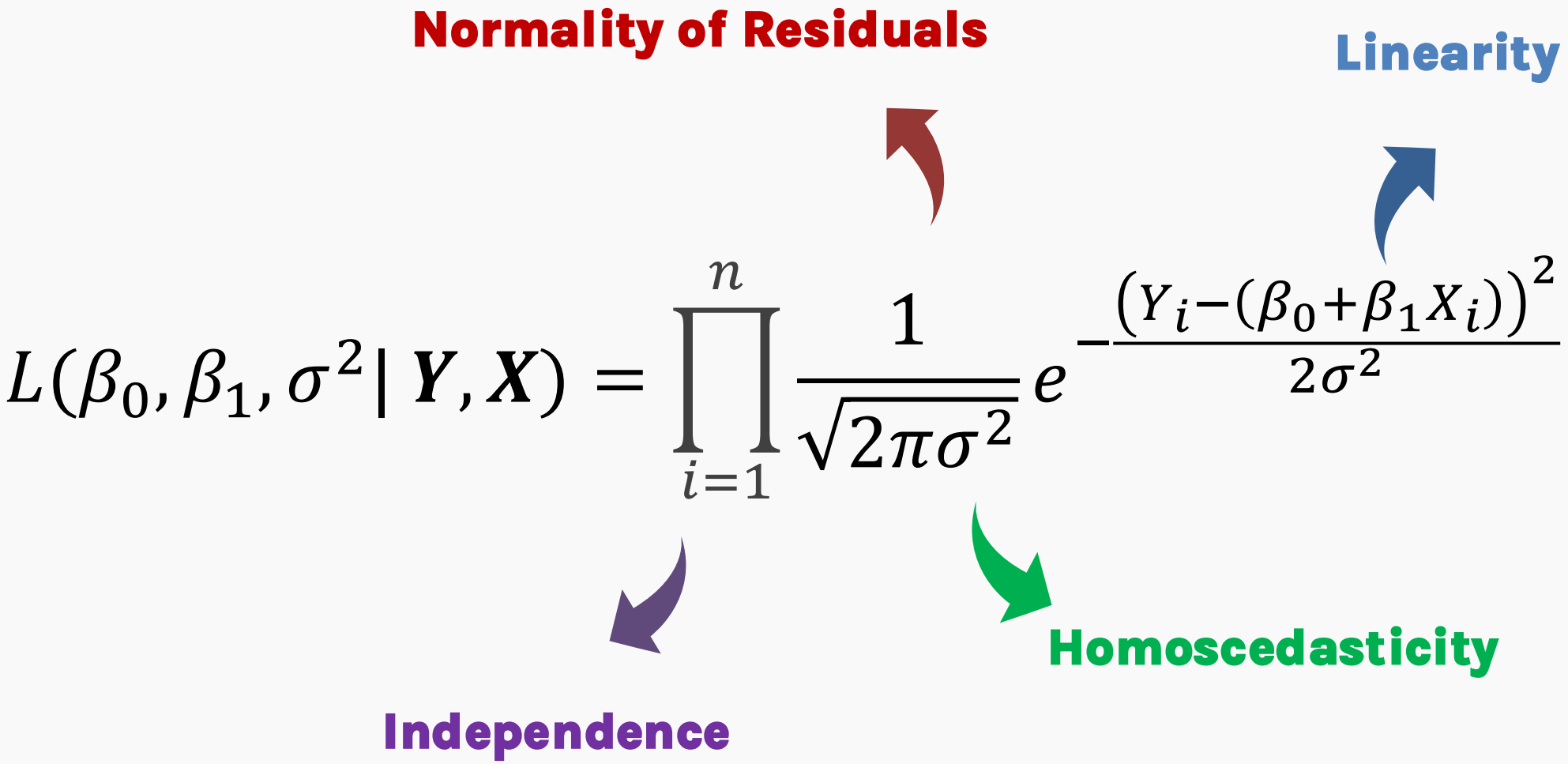
$$L(\beta_0, \beta_1, \sigma^2 | Y_i, X_i) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}\left(\frac{Y_i - (\beta_0 + \beta_1 X_i)}{\sigma}\right)^2}$$

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

The joint likelihood function for this probability model becomes:

$$L(\beta_0, \beta_1, \sigma^2 | \boldsymbol{Y}, \boldsymbol{X}) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}\left(\frac{Y_i - (\beta_0 + \beta_1 X_i)}{\sigma}\right)^2}$$

Assumptions of Linear Regression



The Likelihood of Linear Regression

The joint likelihood function for this probability model becomes:

$$L(\beta_0, \beta_1, \sigma^2 | \boldsymbol{Y}, \boldsymbol{X}) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}\left(\frac{Y_i - (\beta_0 + \beta_1 X_i)}{\sigma}\right)^2}$$

Which leads to the log-likelihood:

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

What should we do with this log-likelihood?

The Likelihood of Linear Regression

Instead of maximizing the log-likelihood we can minimize the ***negative-log-likelihood:***

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

Which is equivalent to minimizing

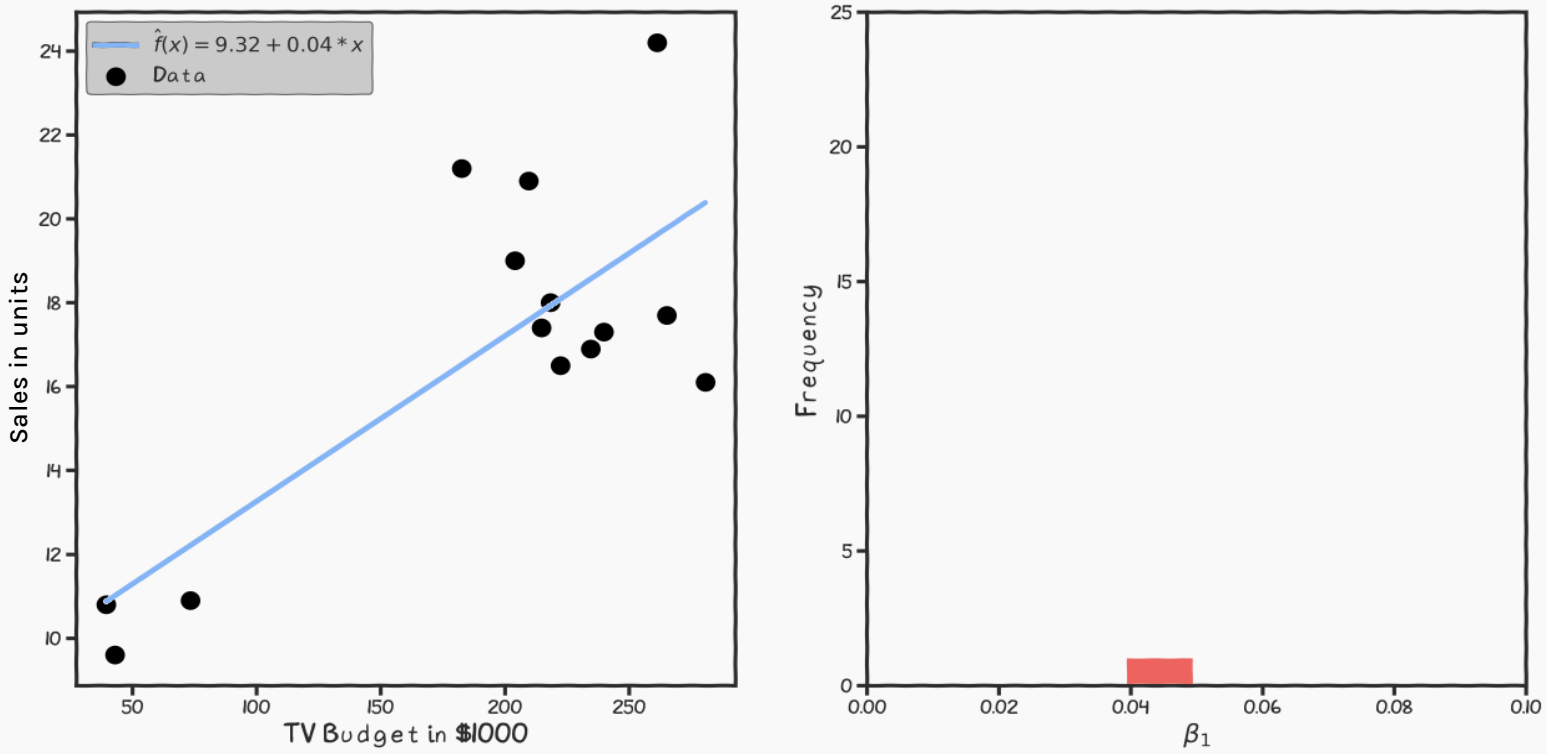
$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - (\beta_0 + \beta_1 x_i))^2$$

Inference:
How well do we know the coefficients?

Confidence intervals for the predictors estimates (cont.)

In our magical realisms, we can now sample multiple times.

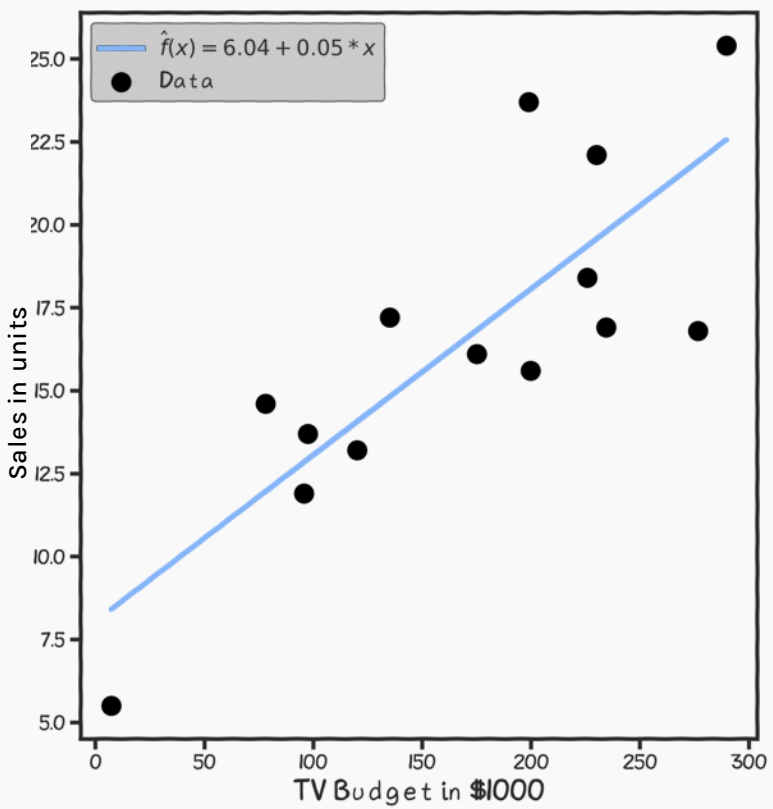
One universe, one sample, one set of estimates for $\hat{\beta}_0, \hat{\beta}_1$

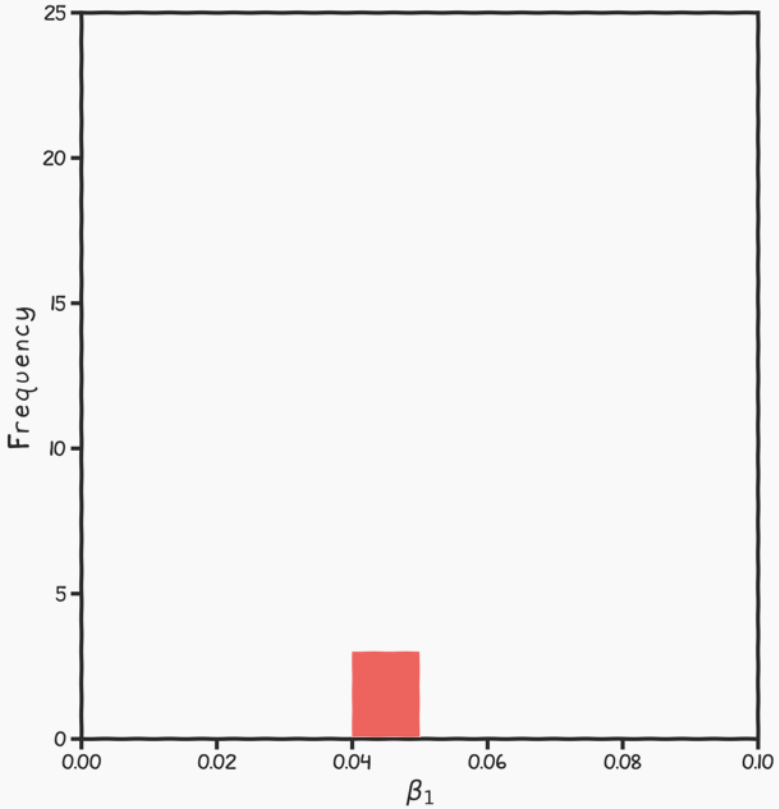


There will be an equivalent plot for $\hat{\beta}_0$ which we don't show here for simplicity

Confidence intervals for the predictors estimates (cont.)

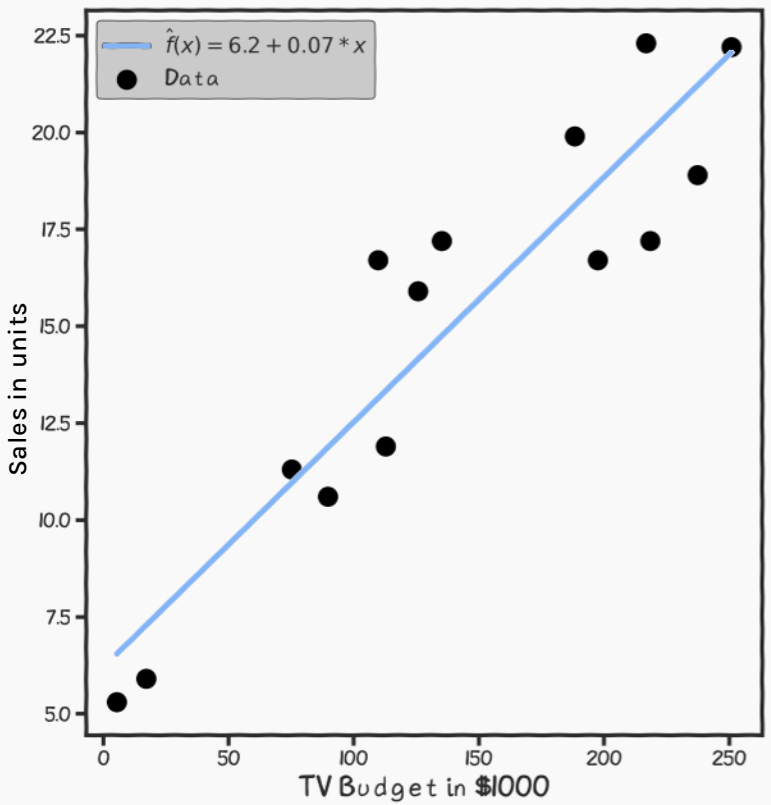
Another sample, another estimate of $\hat{\beta}_0, \hat{\beta}_1$

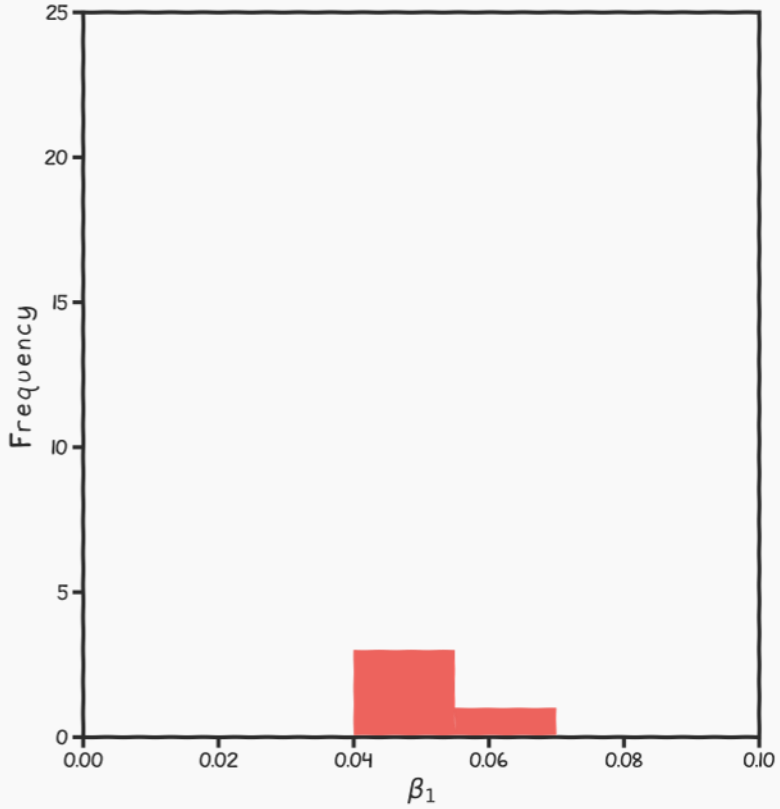




Confidence intervals for the predictors estimates (cont.)

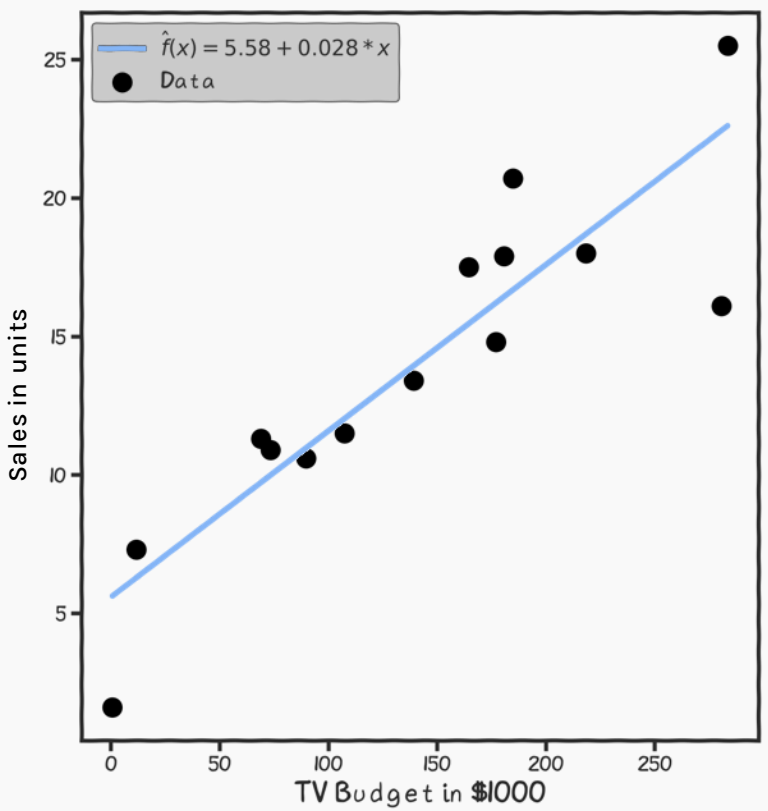
Again

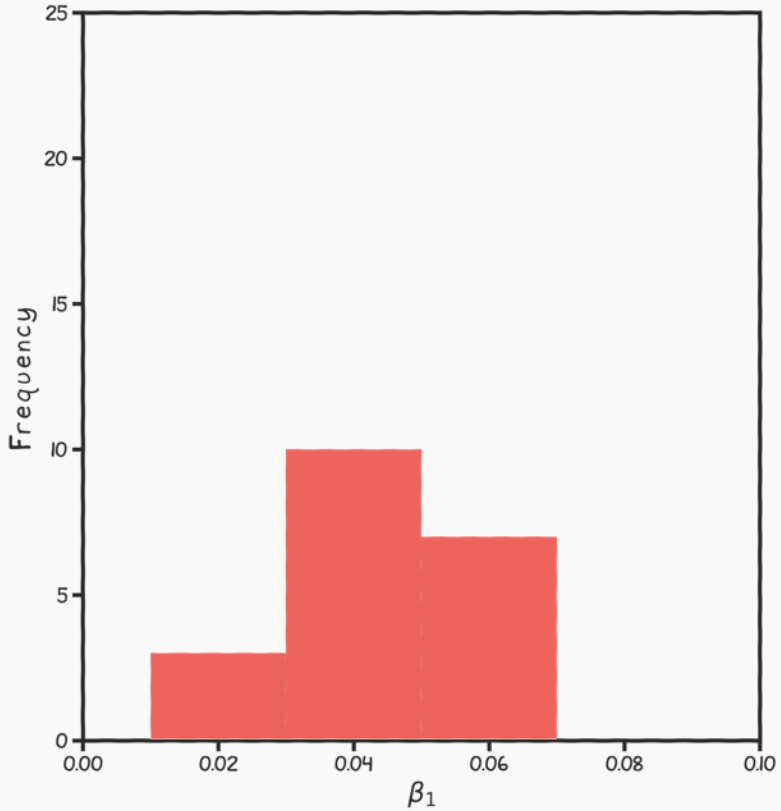




Confidence intervals for the predictors estimates (cont.)

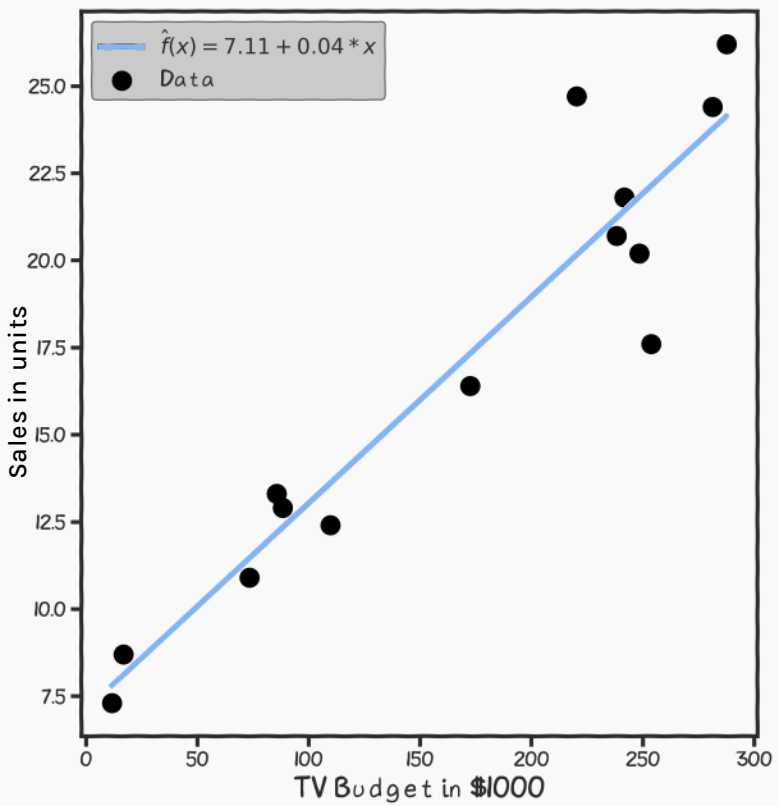
And again

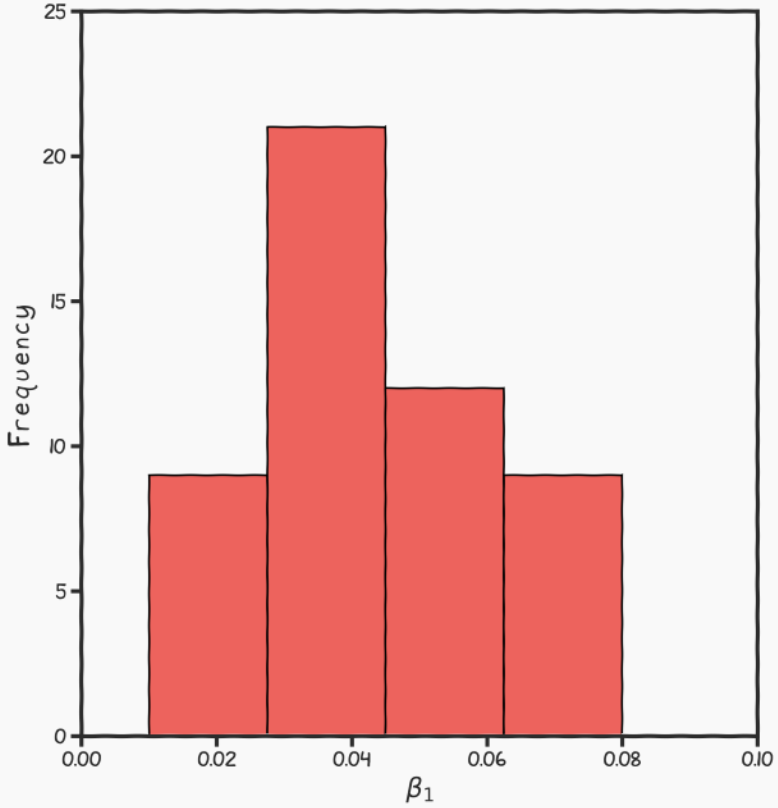




Confidence intervals for the predictors estimates (cont.)

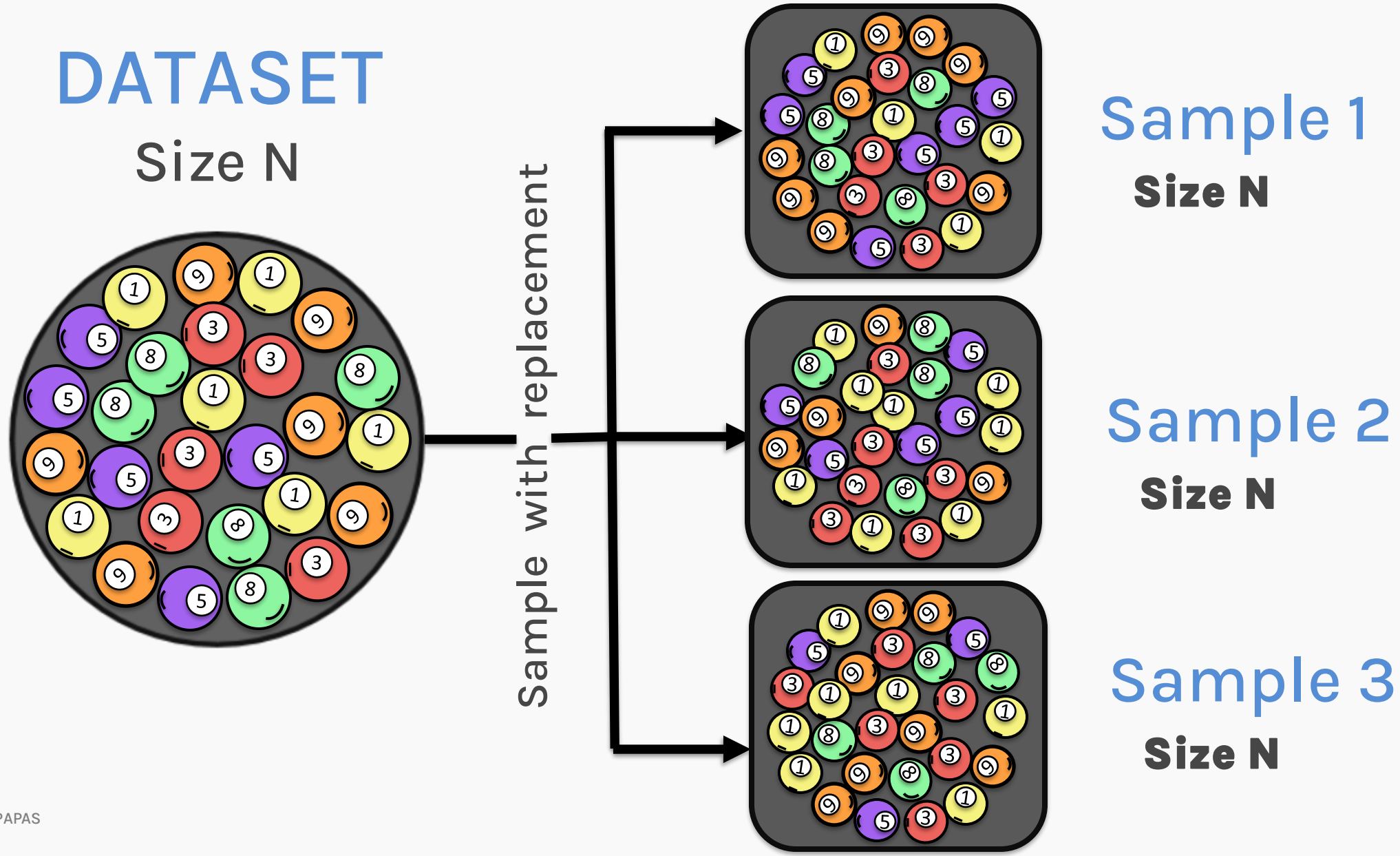
Repeat this for 100 times, until we have enough samples of $\hat{\beta}_0, \hat{\beta}_1$.



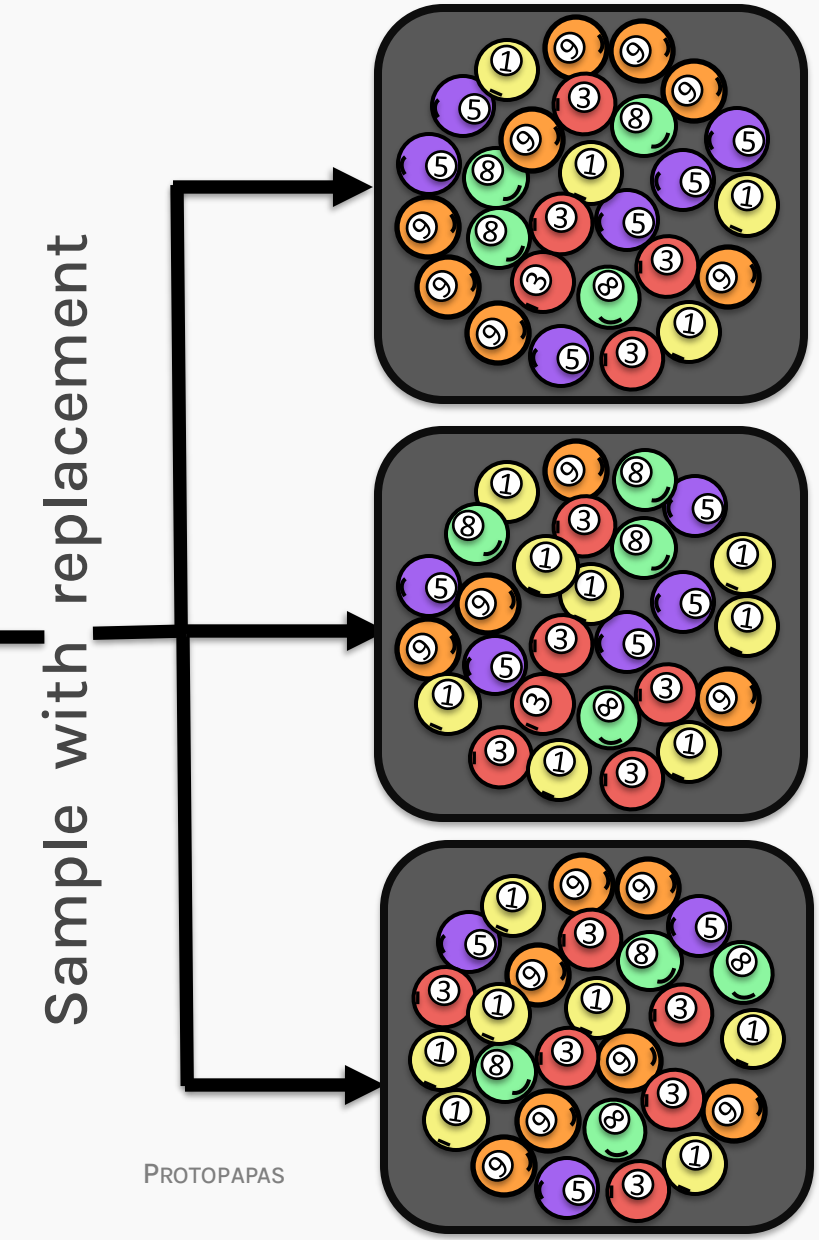


Bootstrap:
Simulating New Datasets

Bootstrap


DATASET
Size N
Sample with replacement
Sample 1
Size N
Sample 2
Size N
Sample 3
Size N

Bootstrap



Sample 1

Size N

Train →

Model 1: $\hat{y} = \hat{\beta}_0^{(1)} + \hat{\beta}_1^{(1)} x$

Sample 2

Size N

Train →

Model 2: $\hat{y} = \hat{\beta}_0^{(2)} + \hat{\beta}_1^{(2)} x$

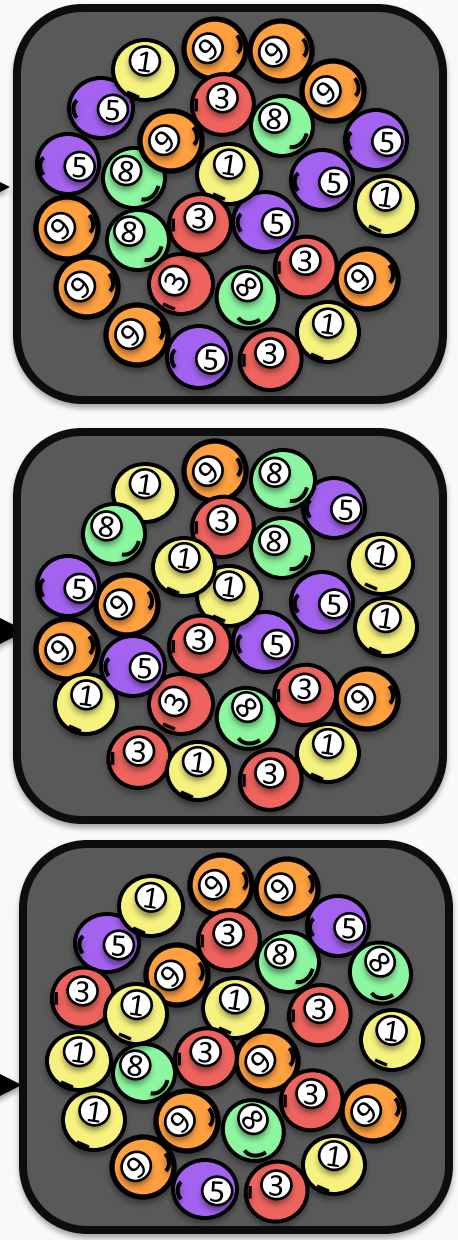
Sample 3

Size N

Train →

Model s: $\hat{y} = \hat{\beta}_0^{(s)} + \hat{\beta}_1^{(s)} x$

Bootstrap



Sample 1 Train → Model 1: $\hat{y} = \hat{\beta}_0^{(1)} + \hat{\beta}_1^{(1)} x$

Size N

Sample 2 Train → Model 2: $\hat{y} = \hat{\beta}_0^{(2)} + \hat{\beta}_1^{(2)} x$

Size N

Sample 3 Train → Model s: $\hat{y} = \hat{\beta}_0^{(s)} + \hat{\beta}_1^{(s)} x$

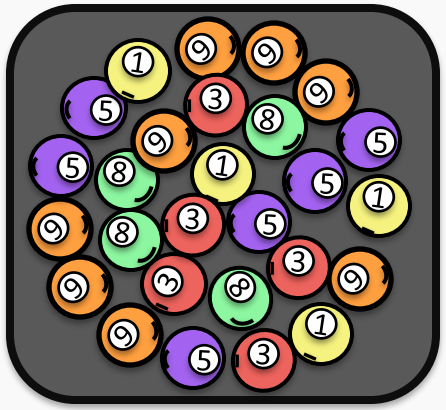
Size N

Combine models

$$\mu_{\hat{\beta}} = \frac{1}{s}\sum_{i=1}^{s} \hat{\beta}^{(i)}$$

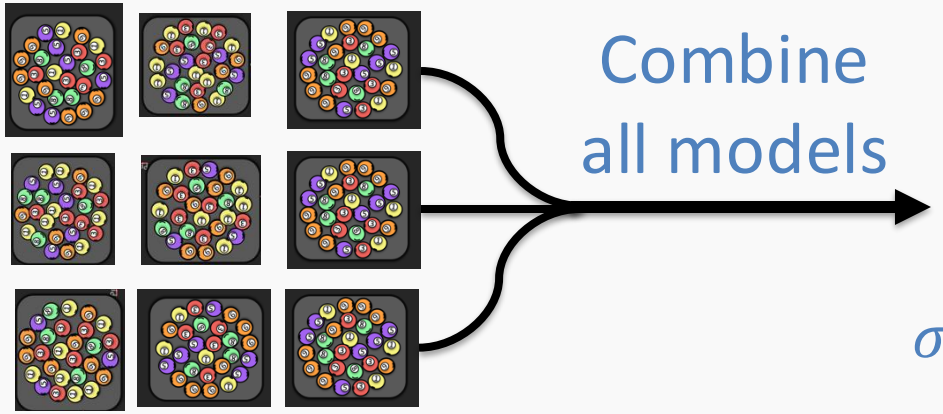
$$\sigma_{\hat{\beta}} = \sqrt{\frac{1}{s-1}\sum_{i=1}^{s} \left(\hat{\beta}^{(i)} - \mu_{\hat{\beta}}\right)^2}$$

In summary, for each "Parallel Universe"…



Train

Model i: $\hat{y} = \hat{\beta}_0^{(i)} + \hat{\beta}_1^{(i)} x$

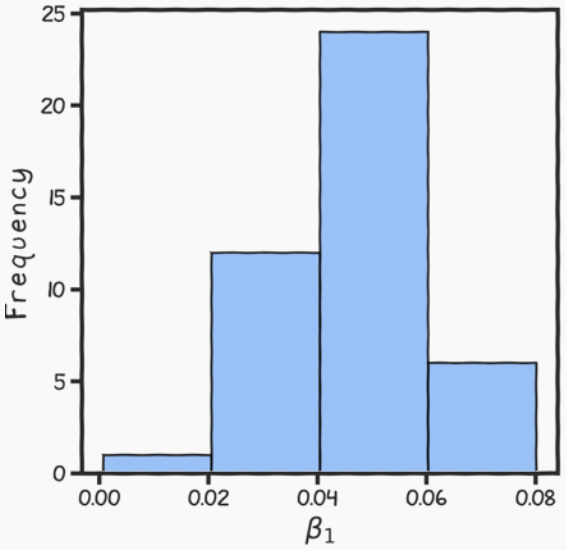


s models

Combine all models

$$\mu_{\hat{\beta}} = \frac{1}{s}\sum_{i=1}^{s} \hat{\beta}^{(i)}$$

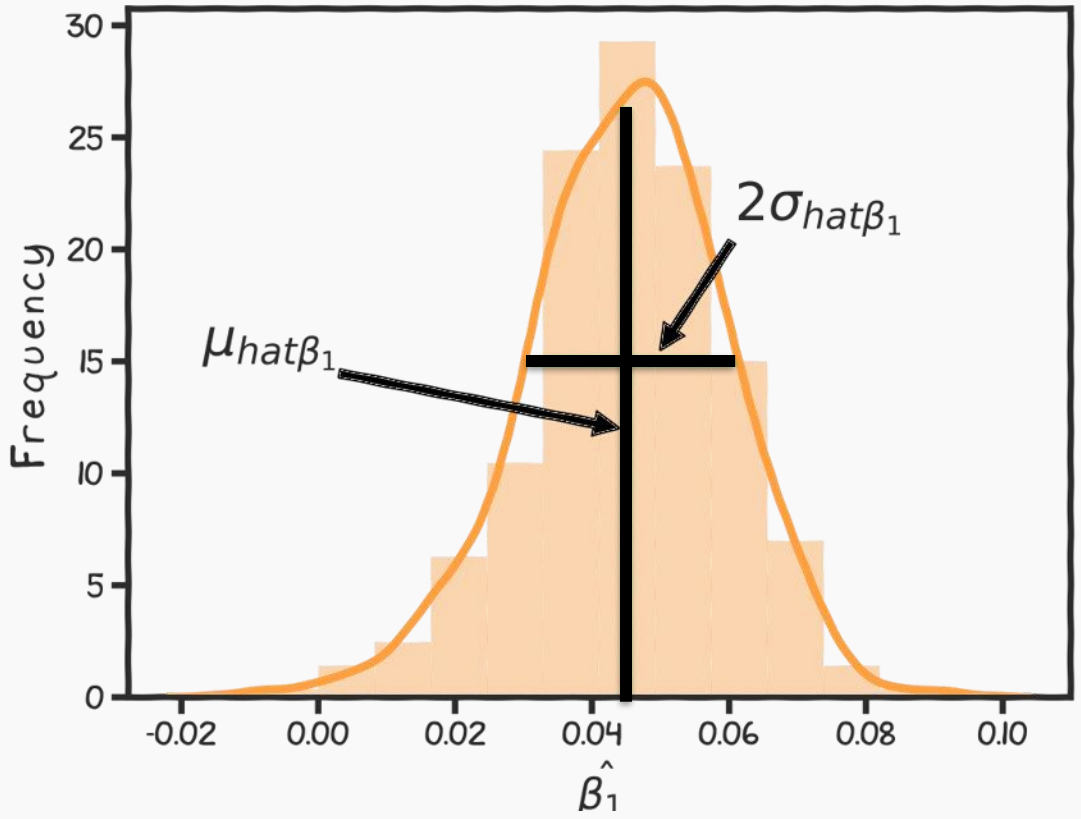
$$\sigma_{\hat{\beta}} = \sqrt{\frac{1}{s-1}\sum_{i=1}^{s} \left(\hat{\beta}^{(i)} - \mu_{\hat{\beta}}\right)^2}$$



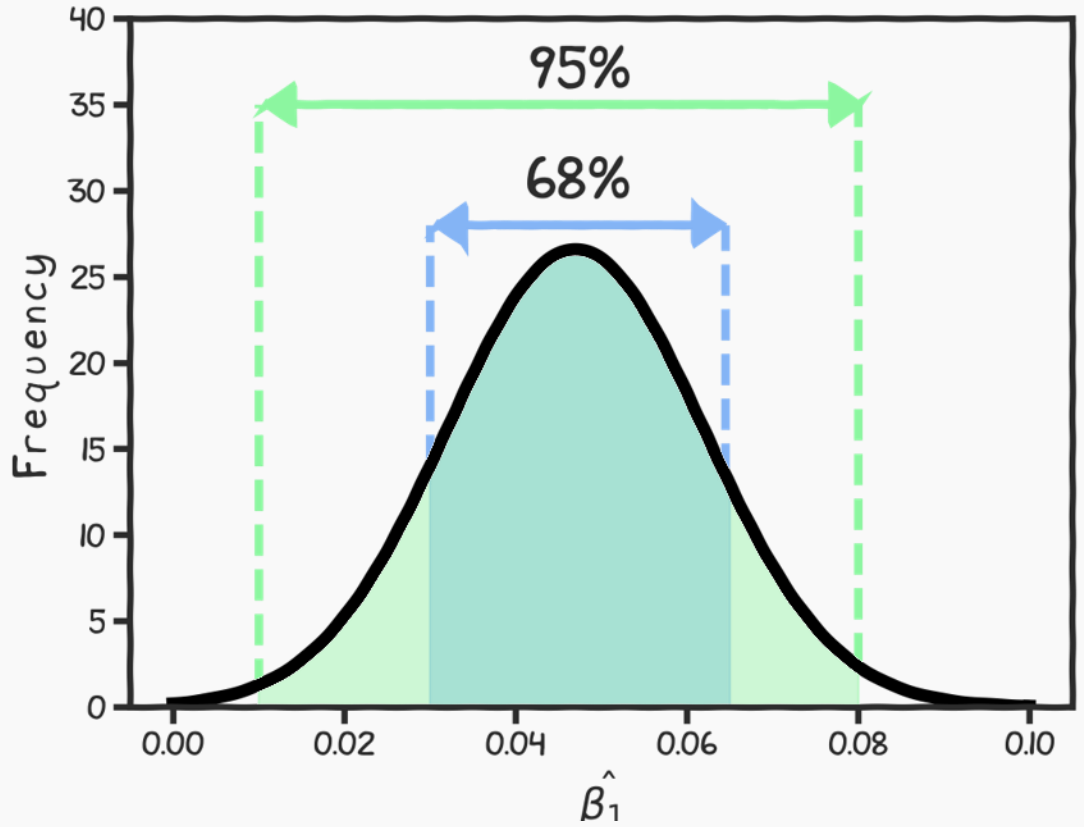
Confidence Intervals & Feature Importance

Confidence intervals for the predictors estimates (cont)

We can empirically estimate the standard deviations $\hat{\sigma}_{\hat{\beta}}$ which are called the **standard errors,** $SE_{\hat{\beta}_0}, SE_{\hat{\beta}_1}$ through bootstrapping.



Confidence intervals for the predictor estimates (cont.)



The standard errors give us a sense of our uncertainty over our estimates.

Typically, we express this uncertainty as a 95% confidence interval, which is the range of values such that the **true** value of β_1 is contained in this interval with 95% percent probability.

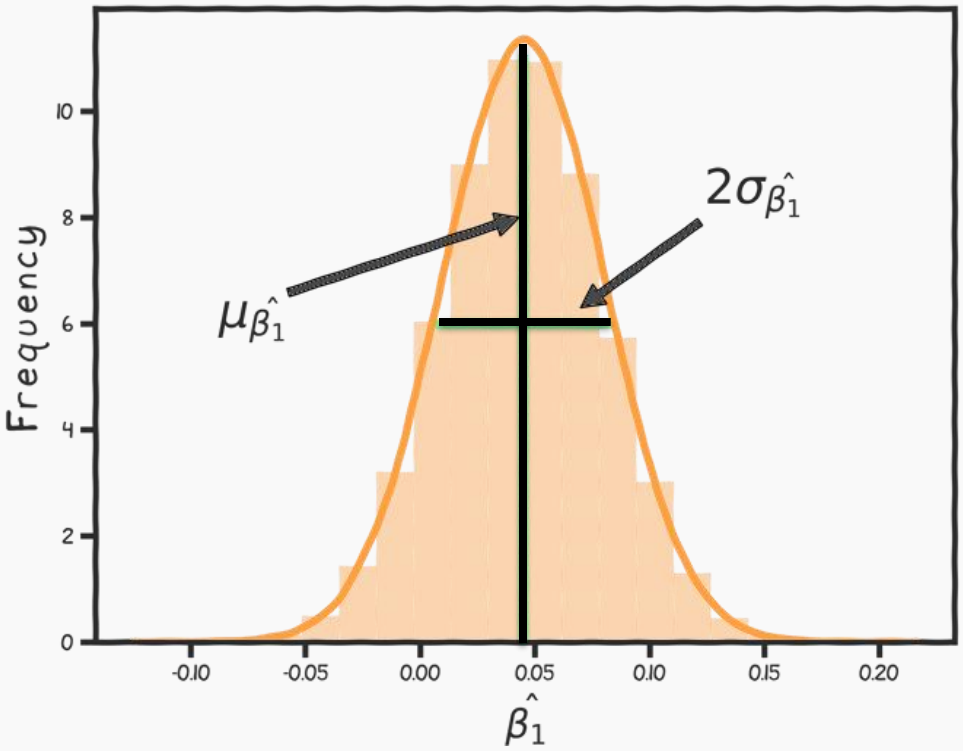
Feature Importance

$$\hat{t}\text{–}\boldsymbol{test} = \frac{\mu_{\hat{\beta}_1}}{\sigma_{\hat{\beta}_1}}$$

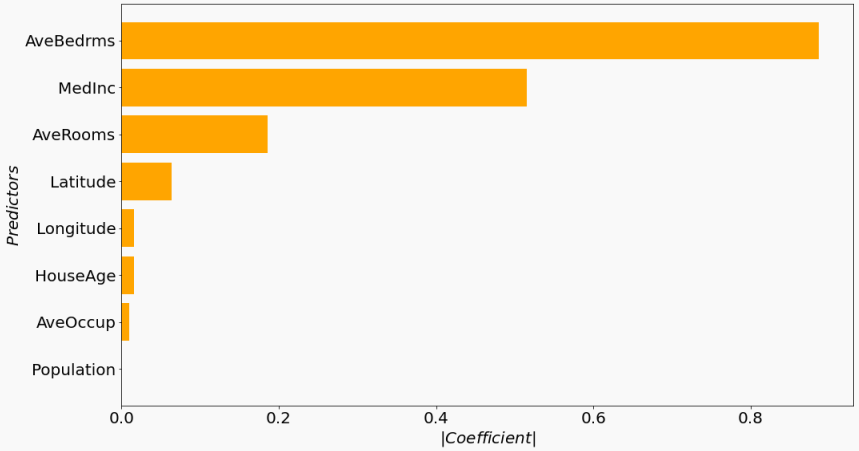
$\hat{t}\text{–}\boldsymbol{test}$ is a scaled version of the usual t-test:

$$t\text{–test} = \frac{\mu_{\hat{\beta}_1}}{\sigma_{\hat{\beta}_1}/\sqrt{n}} = \sqrt{n}\,\hat{t}\text{–test}$$

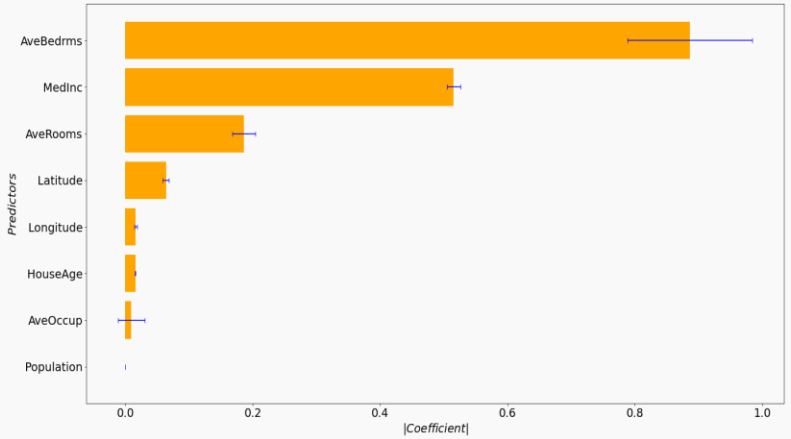
n is the number of bootstraps.



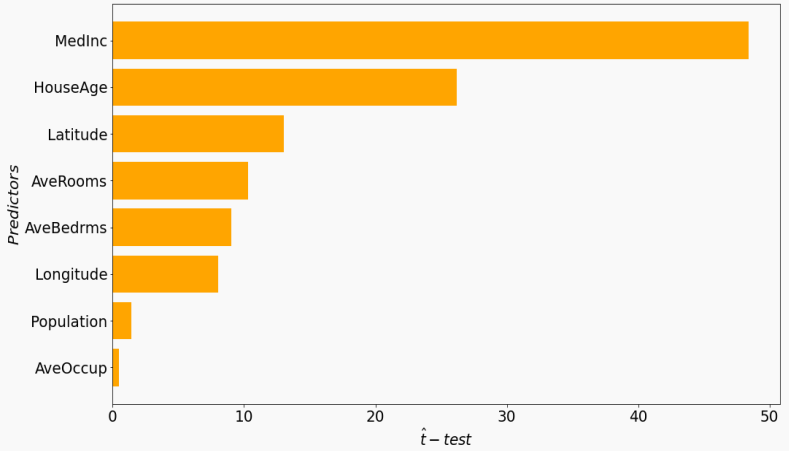
Feature Importance



The absolute value of the coefficients.



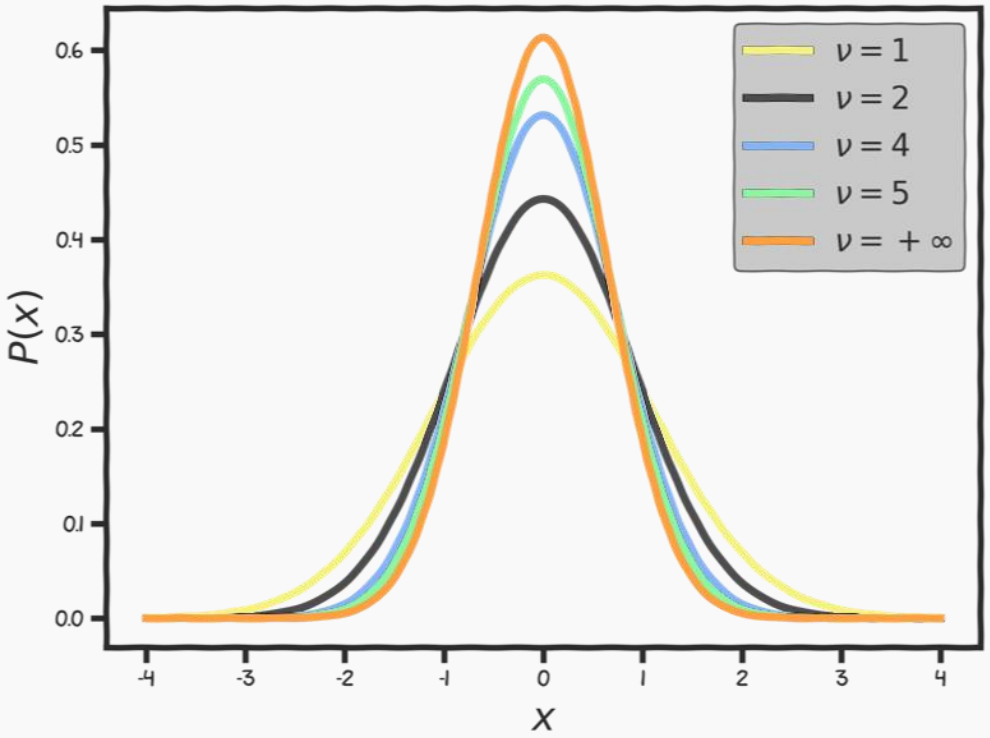
The absolute value of the coefficients over multiple bootstraps and includes the uncertainty of the coefficients.



The $\hat{t}$-test. Notice the rank of the importance has changed.

Feature Importance

It turns out we do not have to do this, because this is a known distribution called student-t distribution.

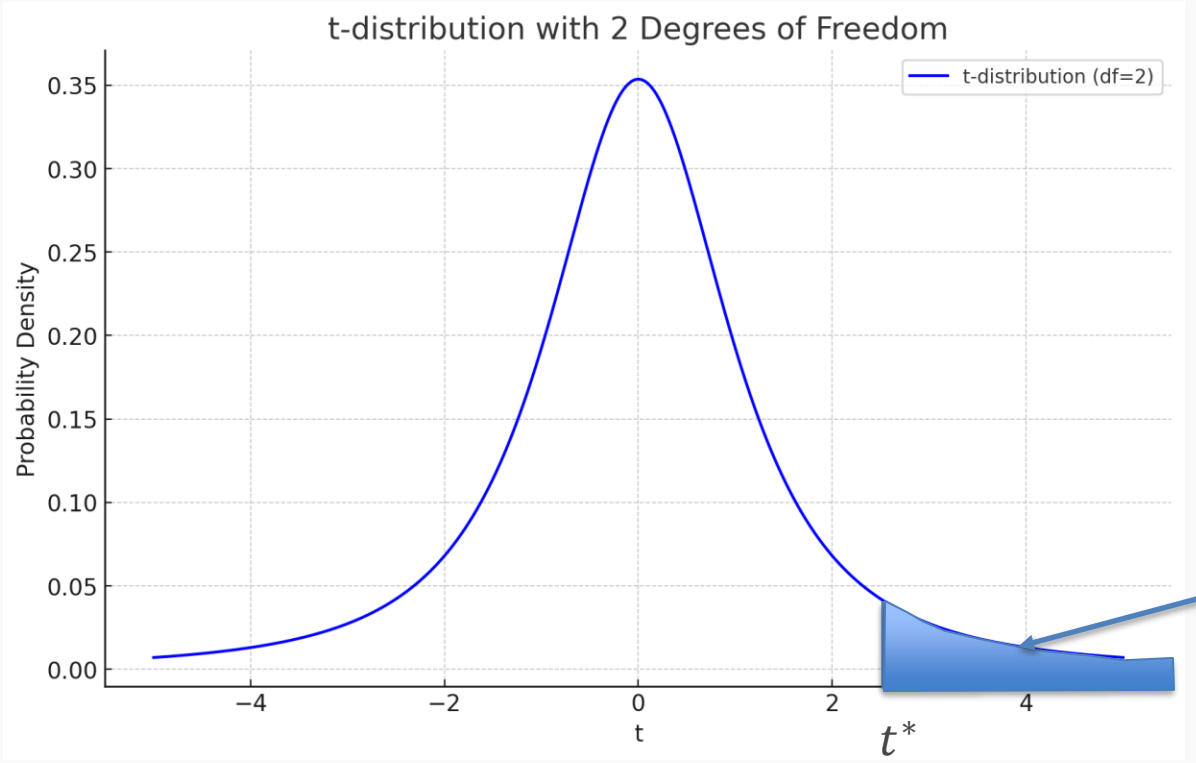


Student-t distribution, where ν is the degrees of freedom (number of data points minus number of predictors) = $\boldsymbol{n} - (\boldsymbol{p} + 1)$).

More about student-t and degrees of freedom

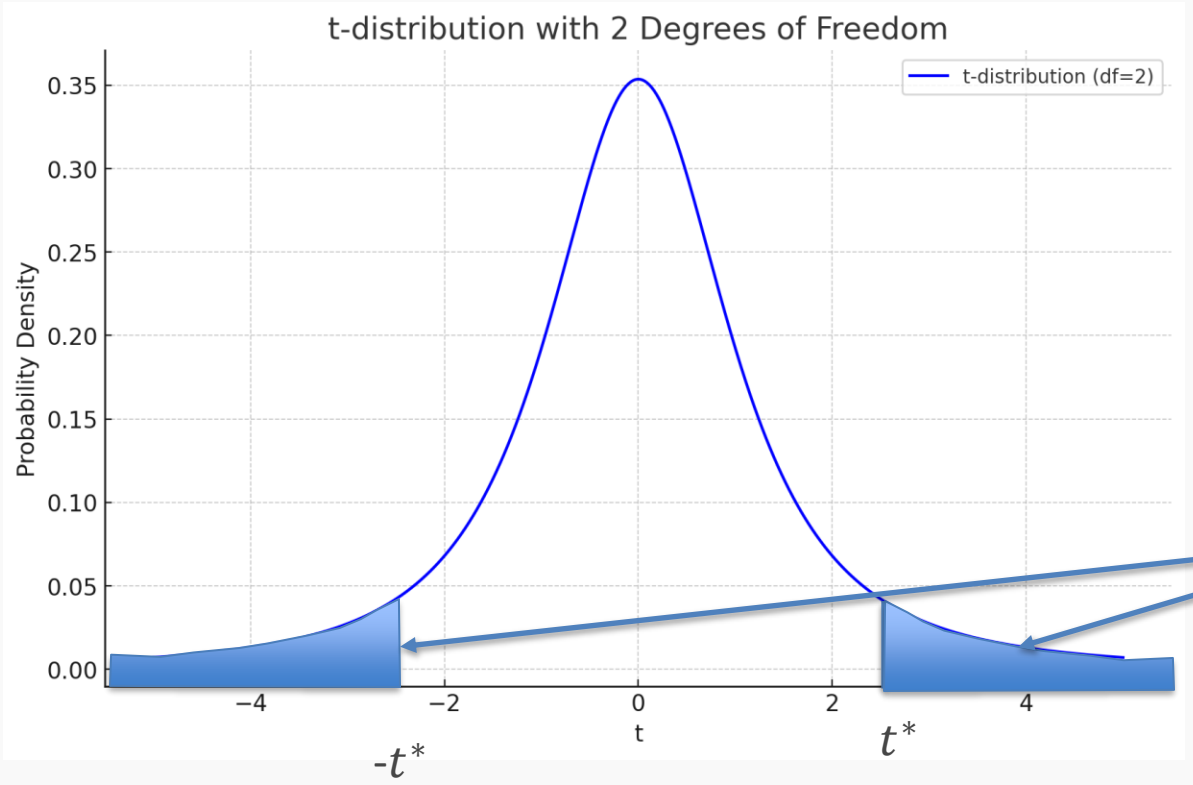
P-value

To compare the t-test values of the predictors from our model, t^*, with the t-tests calculated using random data, t^R, we estimate the probability of observing $t^R \geq t^*$.



P-value

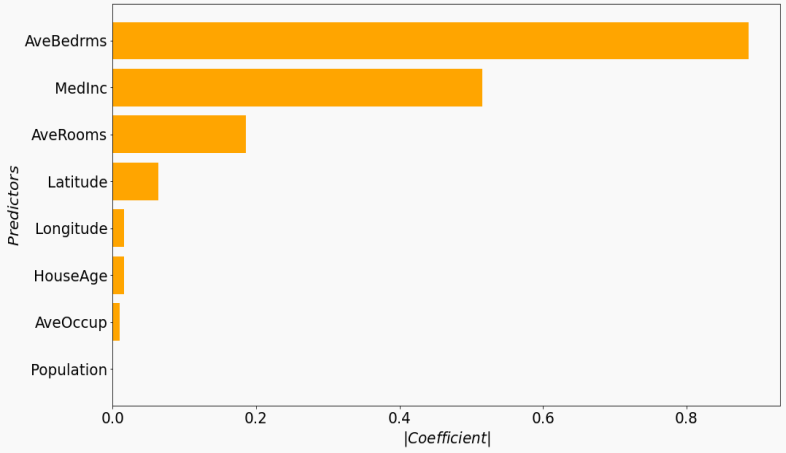
Actually, we need compare $|t^*|$, with the t-tests calculated using random data, $|t^R|$, we estimate the probability of observing $|t^R| \geq |t^*|$.

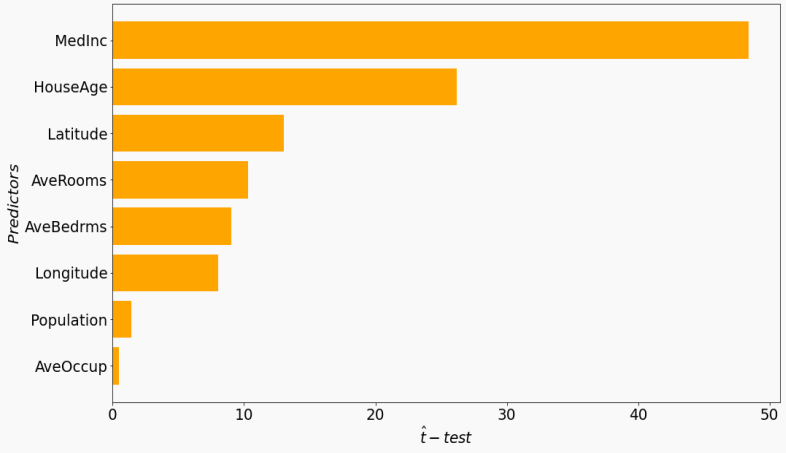


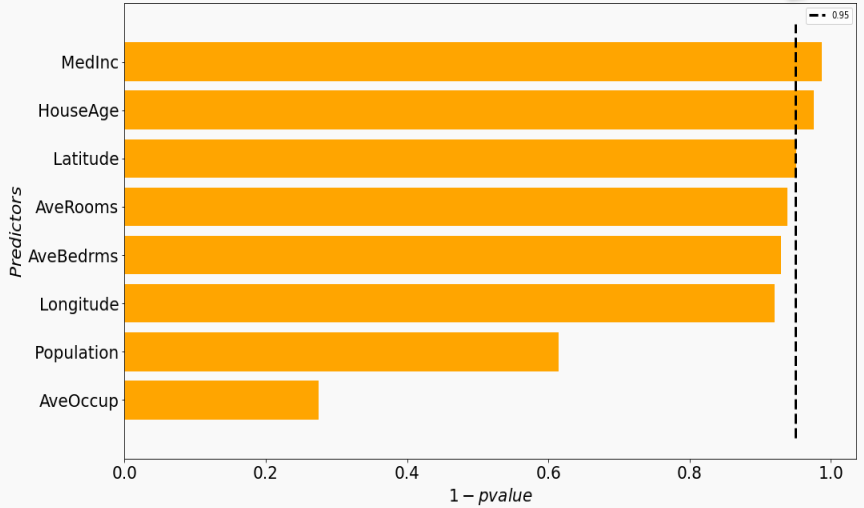
$P(|t^R| \geq |t^*|)$

Feature Importance

Any predictor with a $1-p$ value higher than this is considered important.







The absolute value of the coefficients over multiple bootstraps and includes the coefficients' uncertainty.

The $\hat{t}$-test. Notice the rank of the importance has changed.

Using the the p-value we also have which predictors are important. Note here we use 1-p.

Hypothesis Testing

1. State Hypothesis:

Null hypothesis:

H_0: There is no relation between X_j and Y, ($\beta_j = 0$).

The alternative:

H_a: There is some relation between X_j and Y, ($\beta_j \neq 0$).

2. Choose test statistics

$$\hat{t} - test = \frac{\mu_{\hat{\beta}_1}}{\widehat{SE}_{\hat{\beta}_1}}$$

Hypothesis Testing

3. Sample:

Using bootstrap we can estimate $\hat{\beta}_1's$, and $\mu_{\hat{\beta}_1}$ and $\widehat{SE}_{\hat{\beta}_1}$ and the $\hat{t} - test$.

4. **Reject or not reject the hypothesis:**

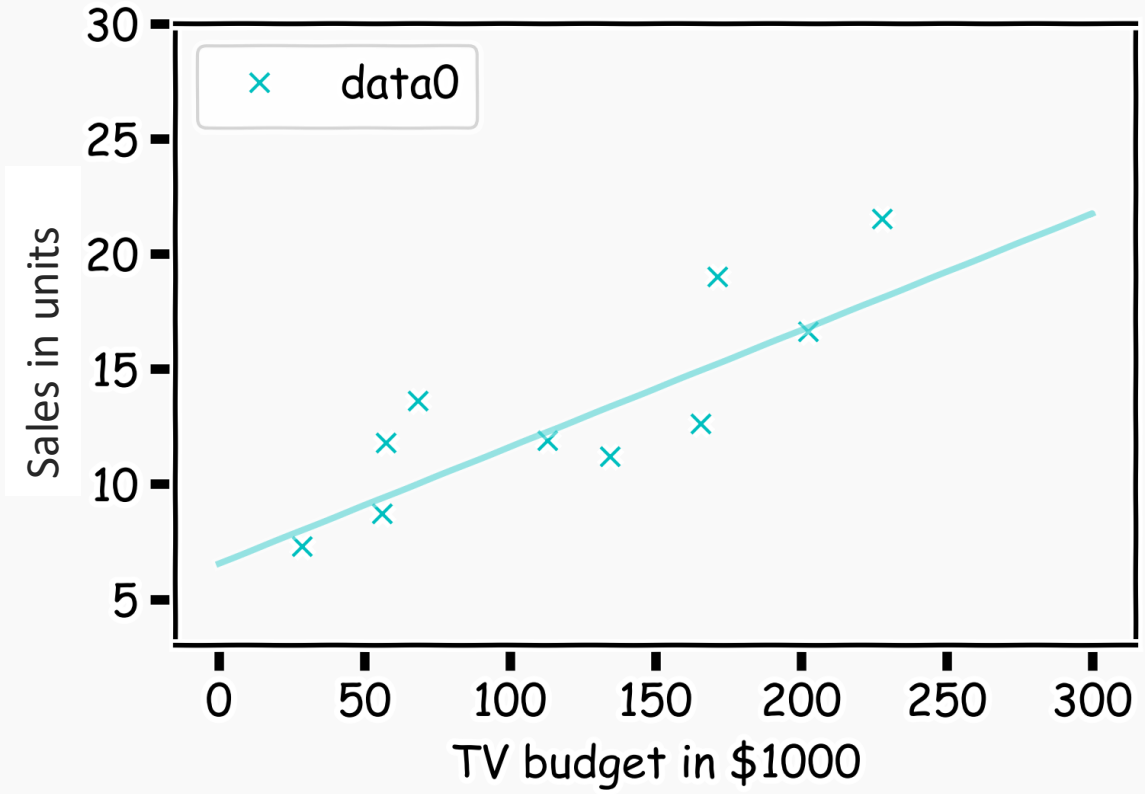
We compute ***p-value*** **,** the probability of observing any value equal to $|t|$ or larger, from random data.

If p-value < p-value-threshold we **reject the null**.

Model CIs & Prediction Intervals

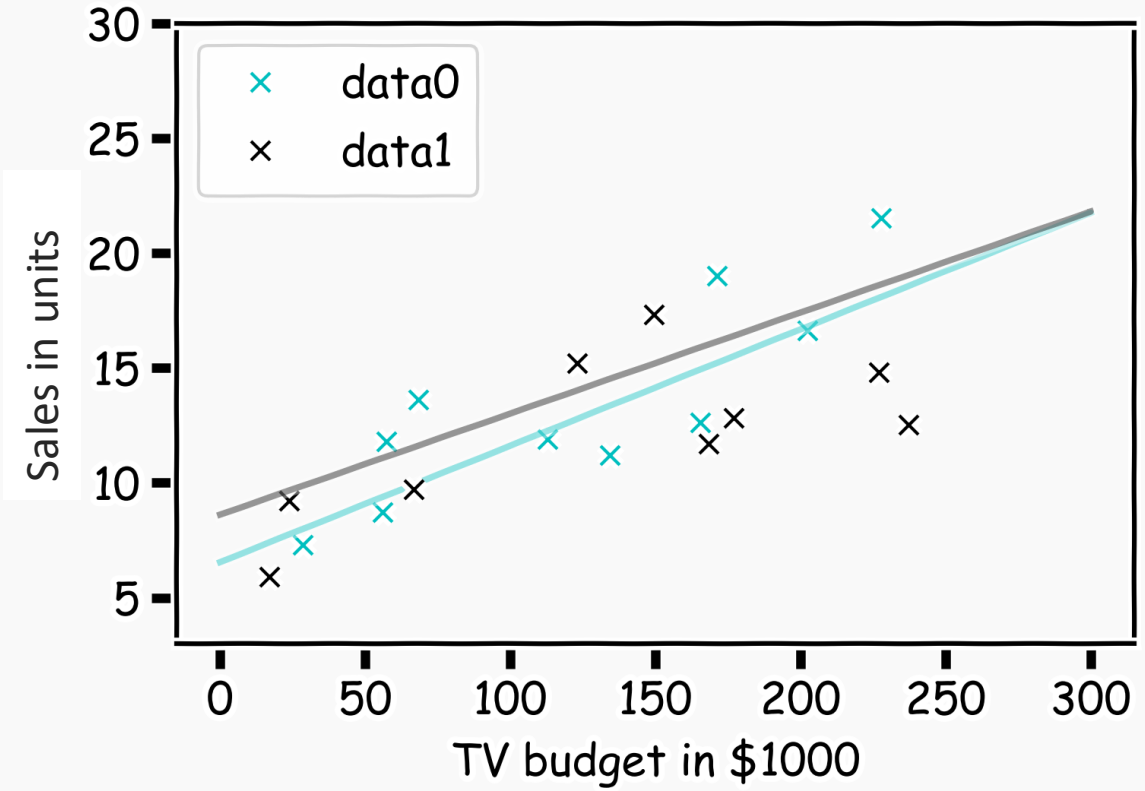
How well do we know $\hat{f}$?

Our confidence in f is directly connected with our confidence in βs. For each bootstrap sample, we have one β, which we can use to determine the model, $f(x) = X\beta$.



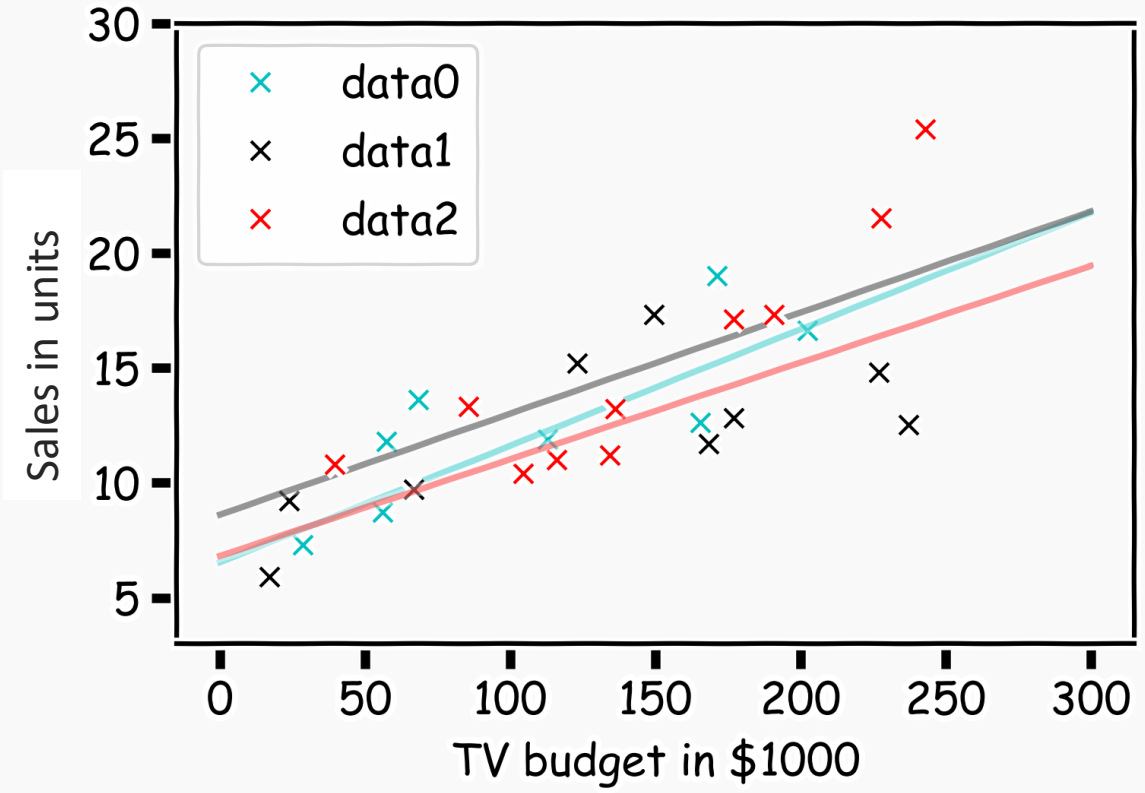
How well do we know $\hat{f}$?

Here we show two different models' predictions given the fitted coefficients, fit on two separate bootstrapped sets of data.



How well do we know $\hat{f}$?

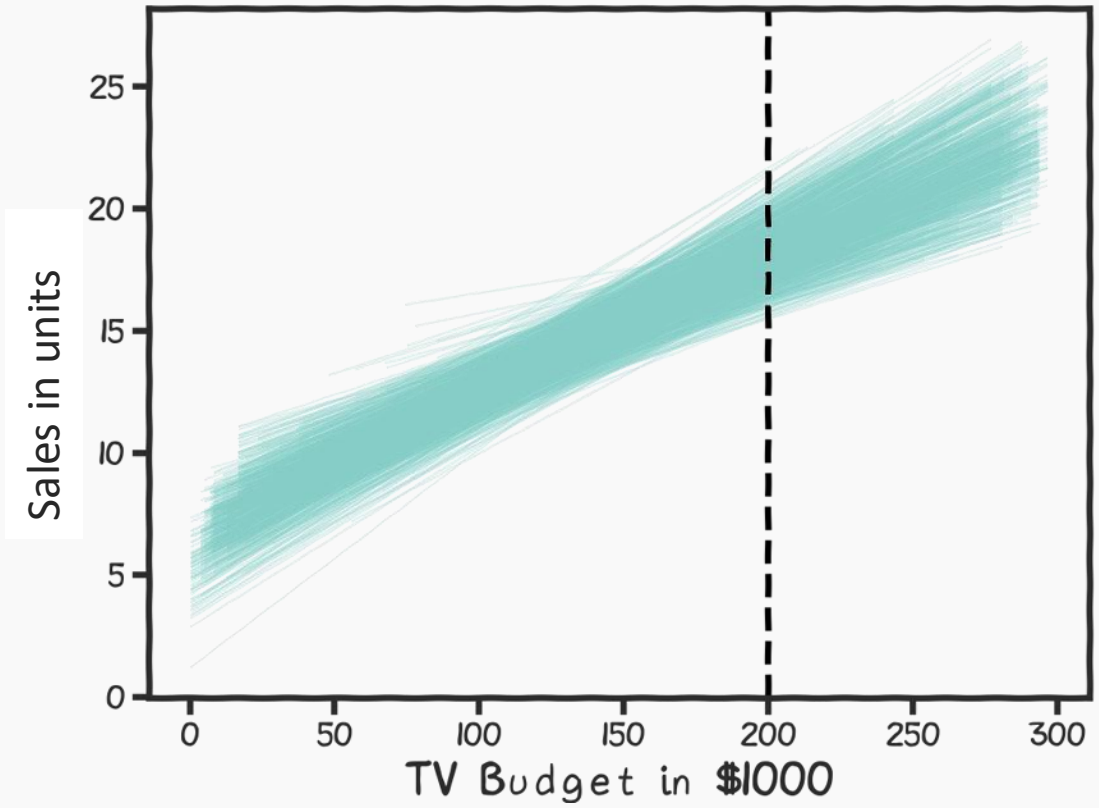
There is one such regression line for every bootstrapped sample.



How well do we know $\hat{f}$?

Below we show all regression lines for a thousand of such bootstrapped samples.

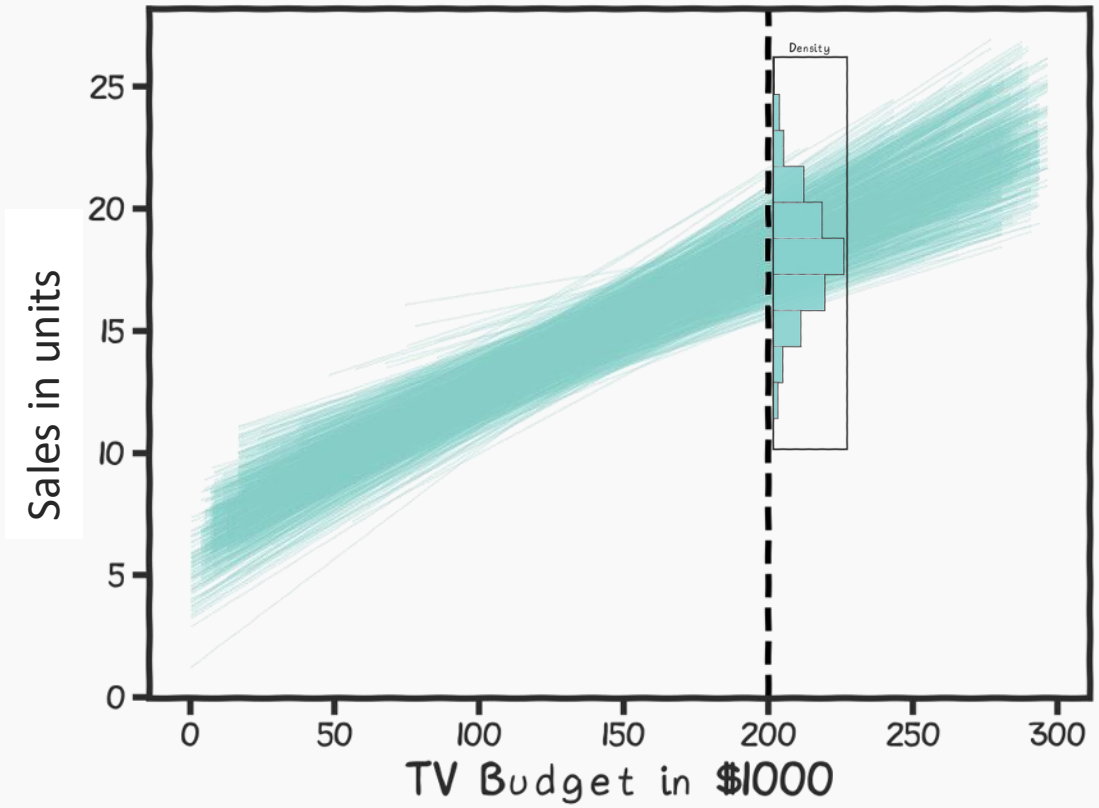
For a given x, we examine the distribution of f and determine the mean and standard deviation (or extract the desired quantiles).



How well do we know $\hat{f}$?

Below we show all regression lines for a thousand of such bootstrapped samples.

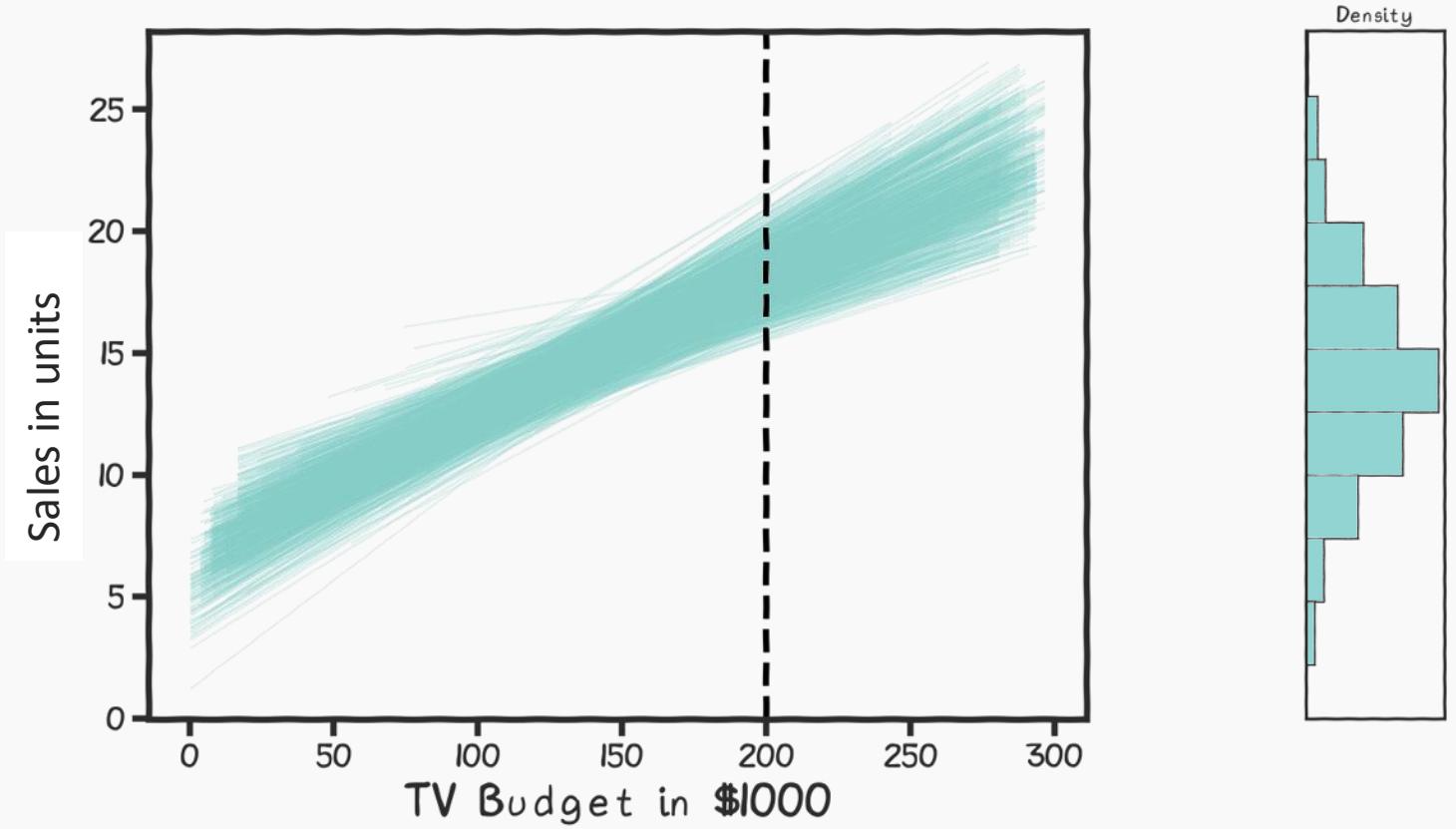
For a given x, we examine the distribution of f and determine the mean and standard deviation (or extract the desired quantiles).



How well do we know $\hat{f}$?

Below we show all regression lines for a thousand of such bootstrapped samples.

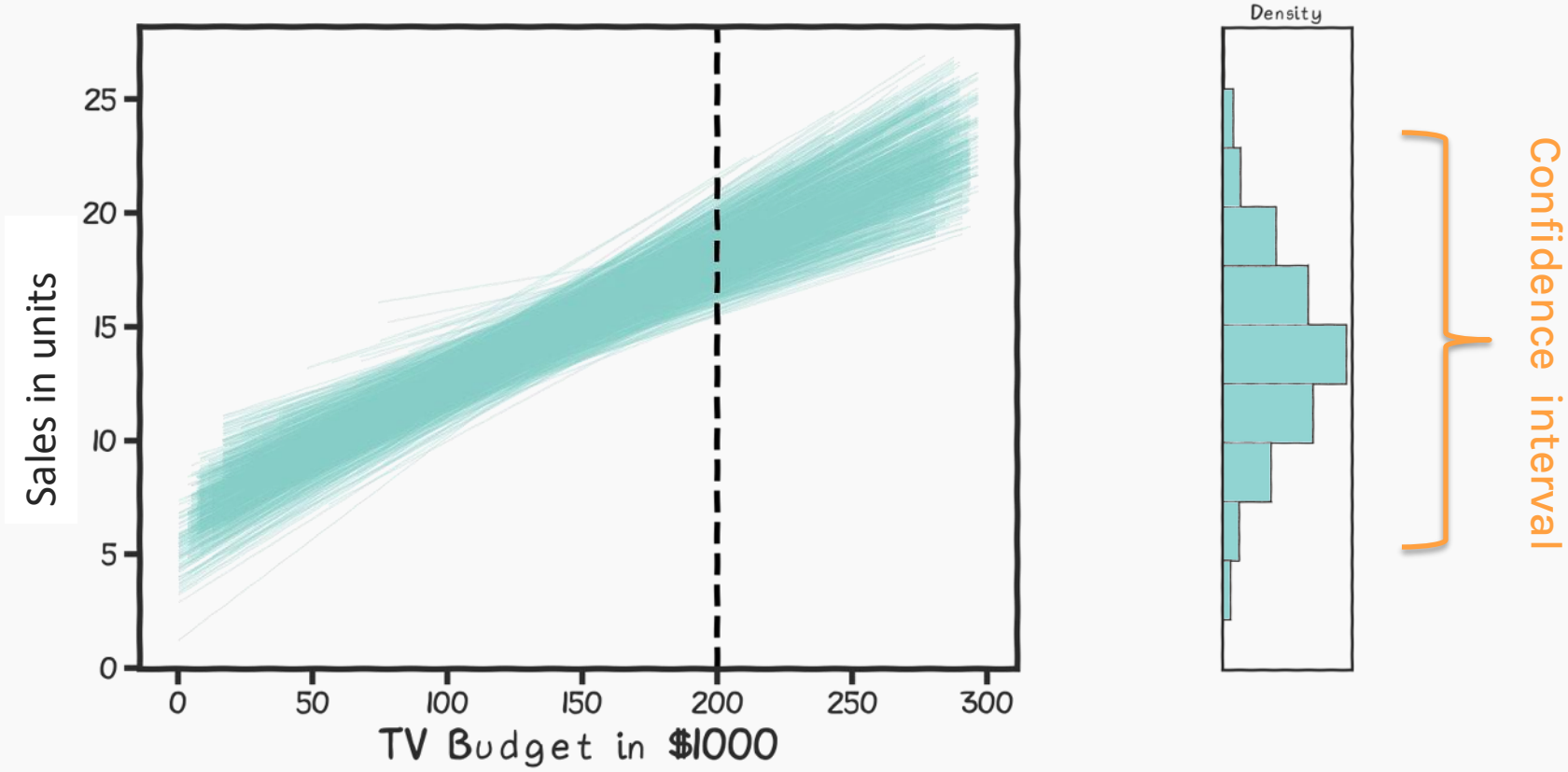
For a given x, we examine the distribution of f and determine the mean and standard deviation (or extract the desired quantiles).



How well do we know $\hat{f}$?

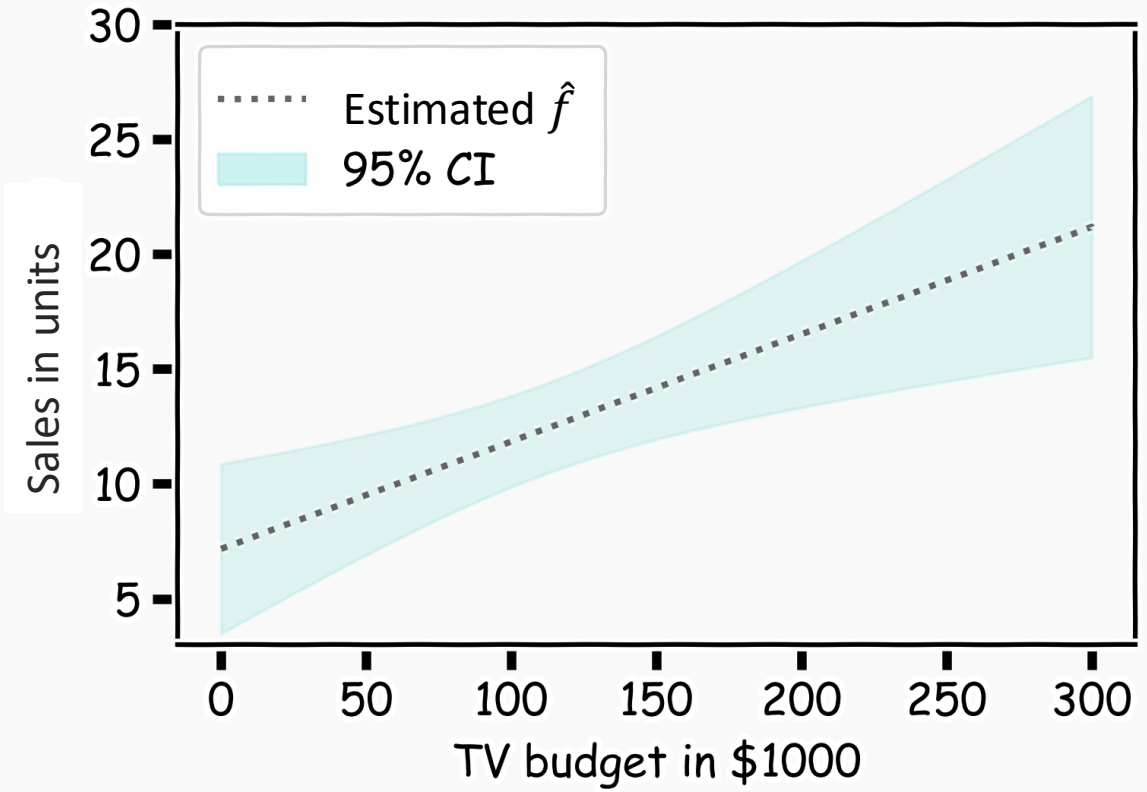
Below we show all regression lines for a thousand of such bootstrapped samples.

For a given x, we examine the distribution of f and determine the mean and standard deviation (or extract the desired quantiles).



How well do we know $\hat{f}$?

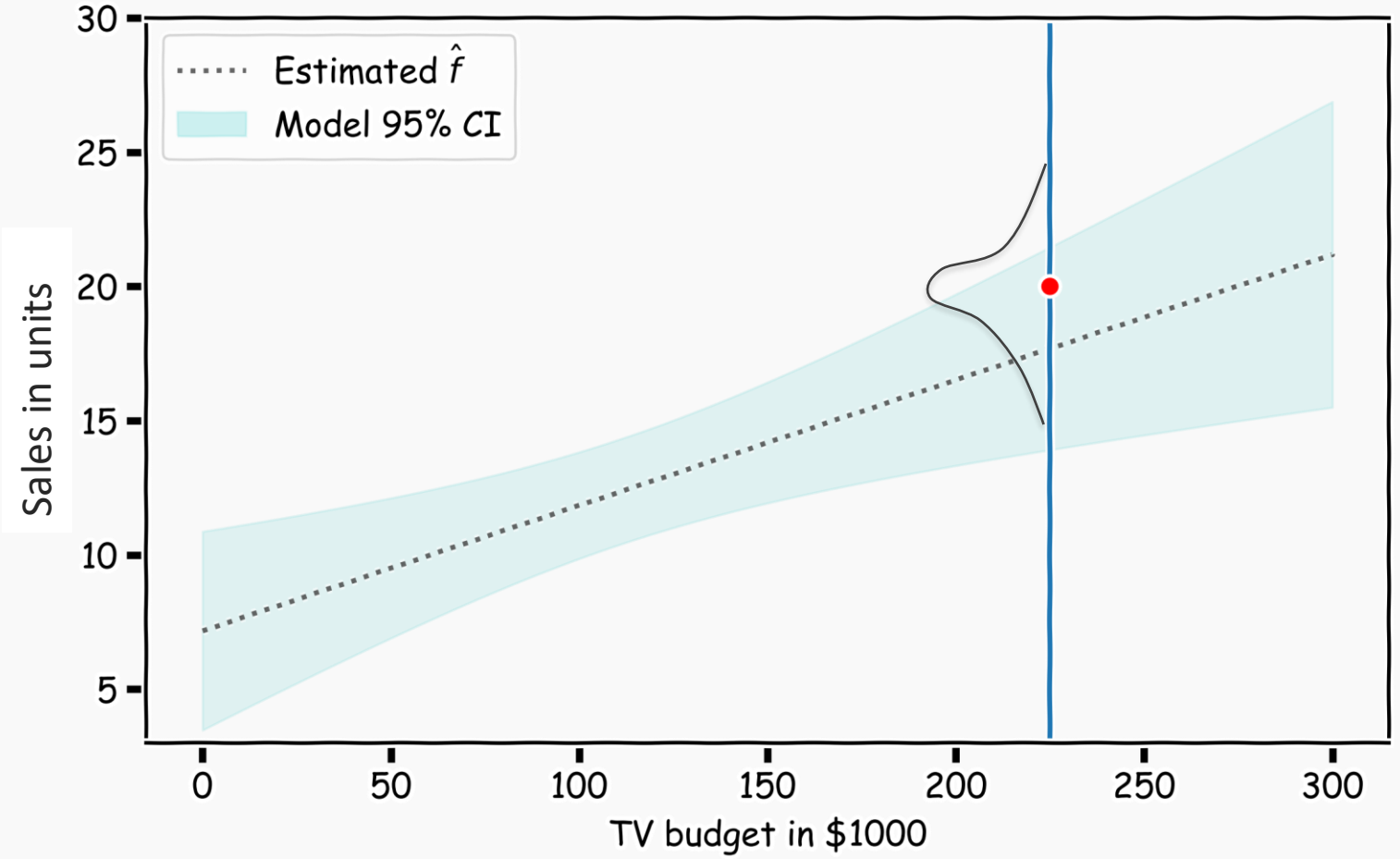
For every x, we calculate the mean of the models, $\widehat{\mu_f}$ (shown with dotted line) and the 95% CI of those models (shaded area).



Confidence in predicting $\hat{y}$

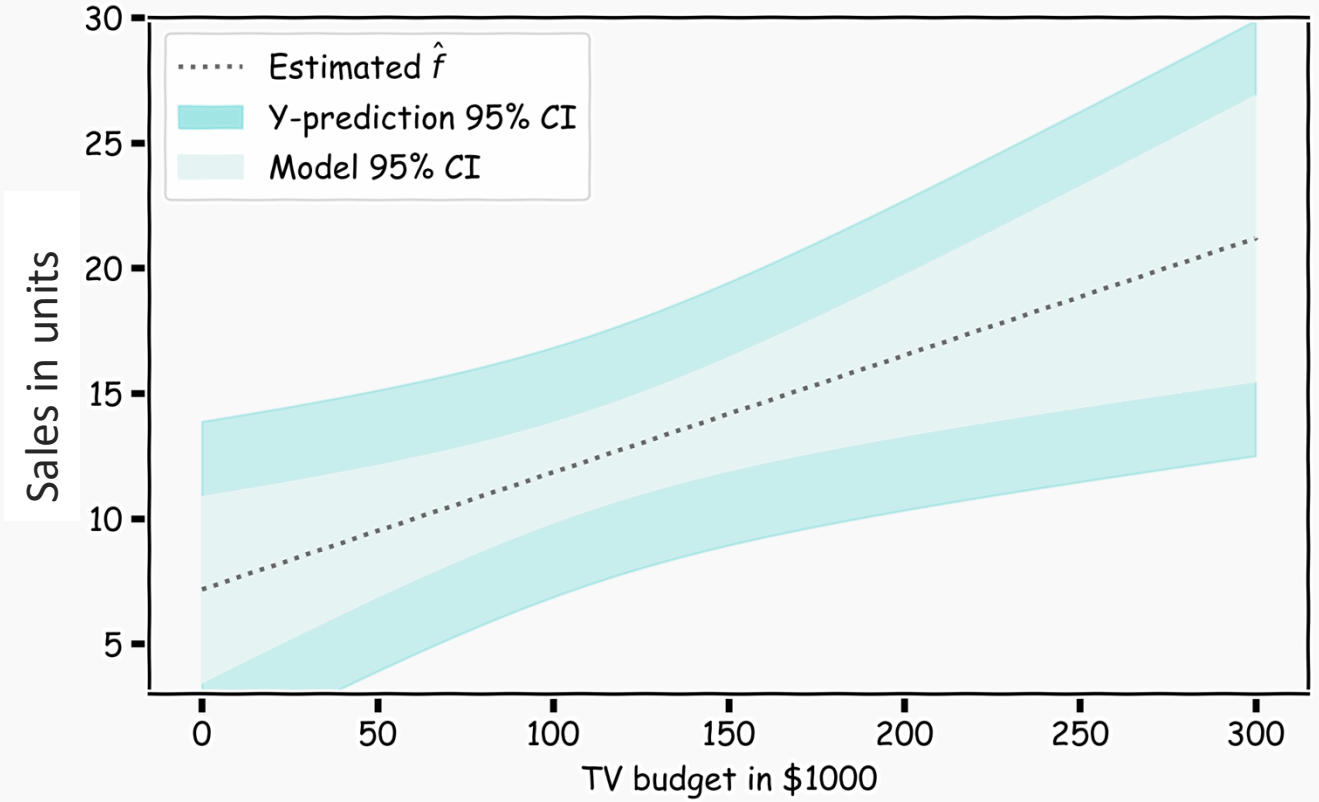
$f(x)$ is a random variable

- For a given x, we have a distribution of models $f(x)$
- For each of these $f(x)$, the prediction for $y \sim N(f(x), \sigma_\epsilon)$



Confidence in predicting $\hat{y}$

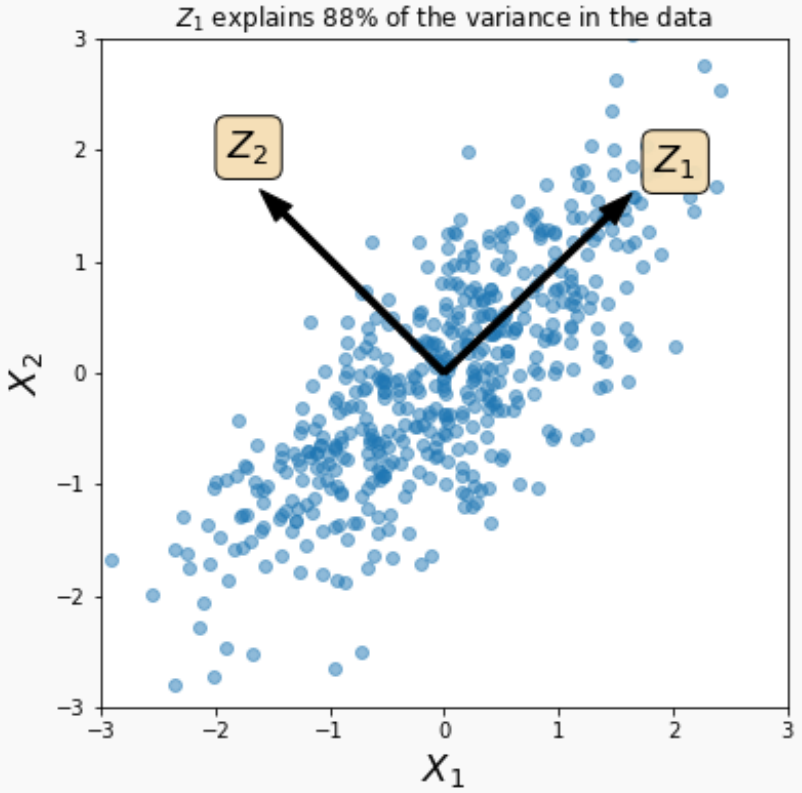
- For a given x, we have a distribution of models $f(x)$
- For each of these $f(x)$, the prediction for $y \sim N(f(x), \sigma_\epsilon)$
- The prediction confidence intervals are then …



PCA

Directions of maximum variance

The vector Z_1 represents the direction of maximum variance in the predictor space. Z_2 is orthogonal and captures the remaining unexplained variance.



If our data were represented in terms of Z_1 and Z_2 then we could decide to keep only the Z_1 component.

This would again cut the dimensionality in half while retaining the most information.

The Math behind PCA

Let $\boldsymbol{X}$ be the n x p matrix with columns X_1,..., X_p (our original predictors), each standardized to have mean zero and variance one, and without the intercept)

Let let $\boldsymbol{W}$ be the p x p matrix whose columns are the eigenvectors of the **covariance matrix**, $\boldsymbol{X}^T\boldsymbol{X}$

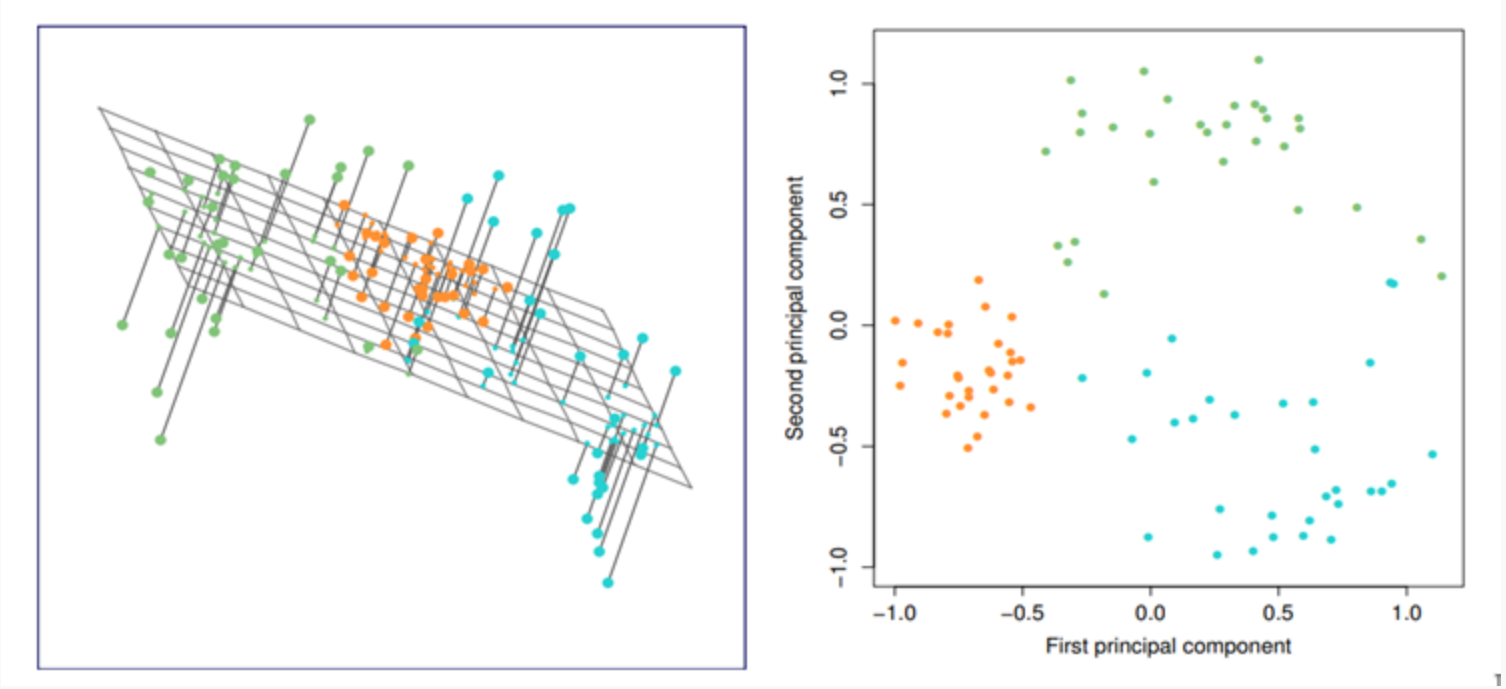
Let $\boldsymbol{Z}$ be the n x p matrix with columns Z_1,..., Z_p (the principal components)

$$\mathbf{Z}_{n\times p} = \mathbf{X}_{n\times p}\mathbf{W}_{p\times p}$$

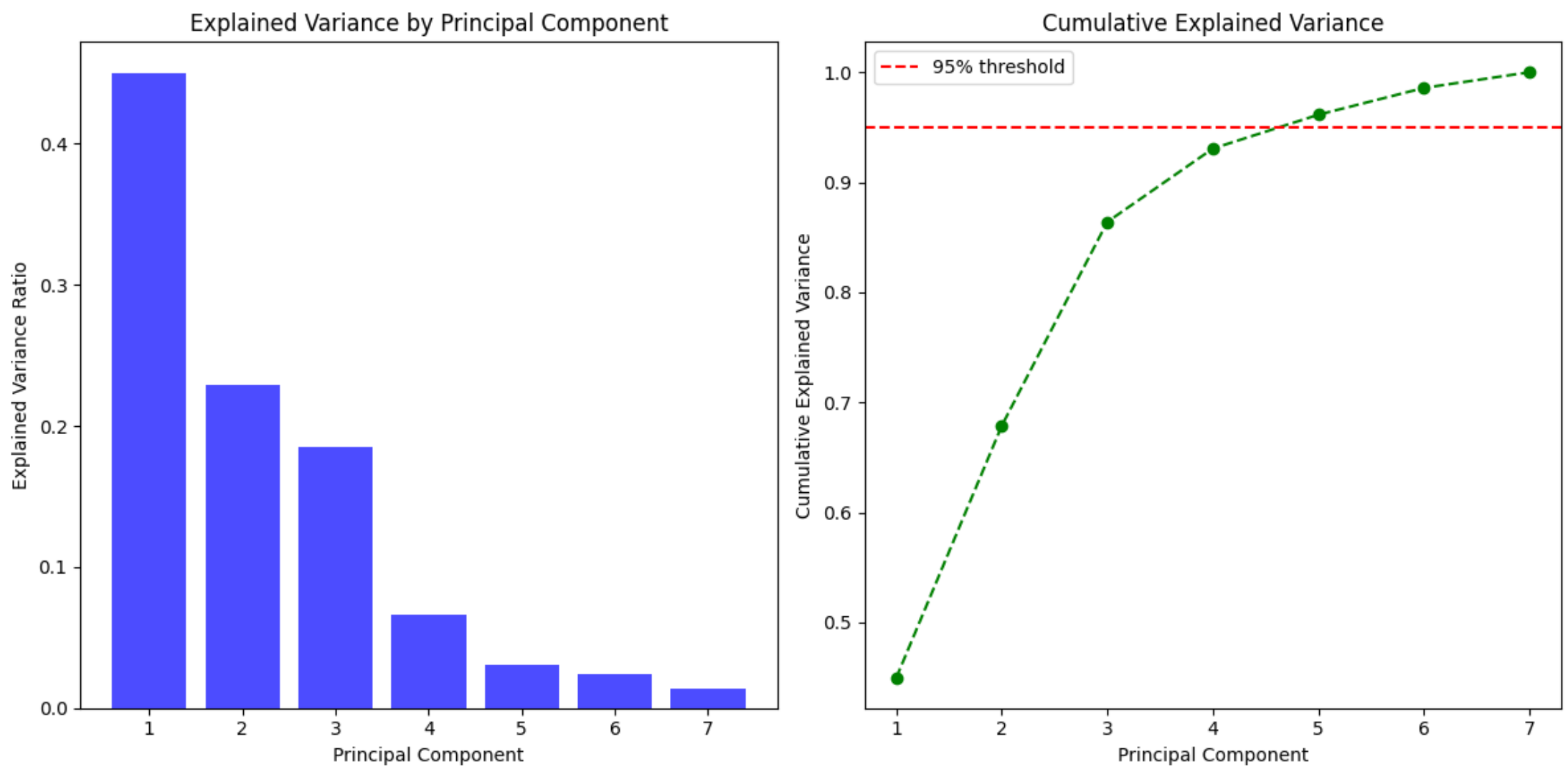
An Alternative Interpretation of PCA

We've seen an interpretation of PCA as finding the directions in the predictor space along which the data varies the most.

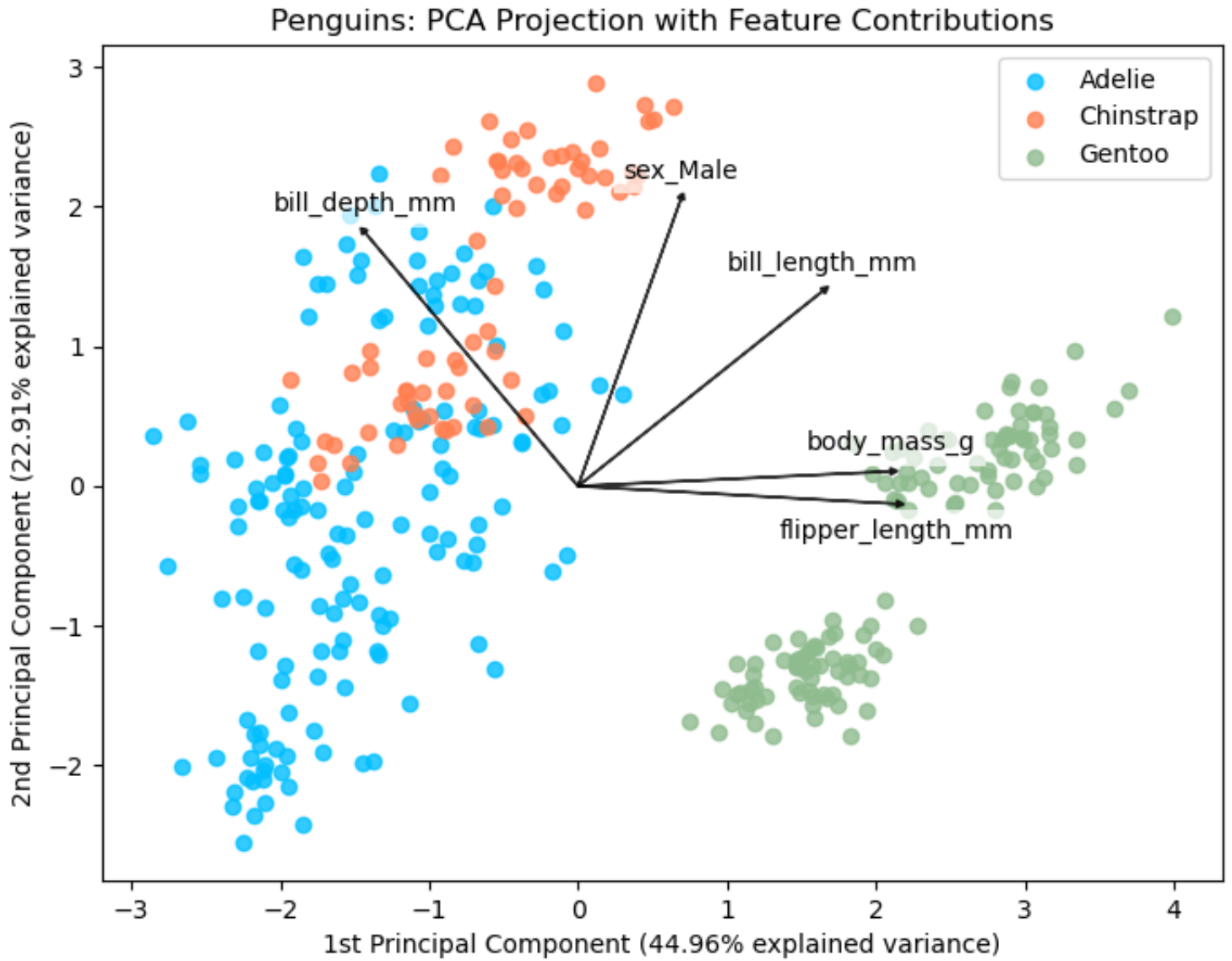
An alternative interpretation is that PCA finds a low-dimensional linear surface which is *closest* to the data points.



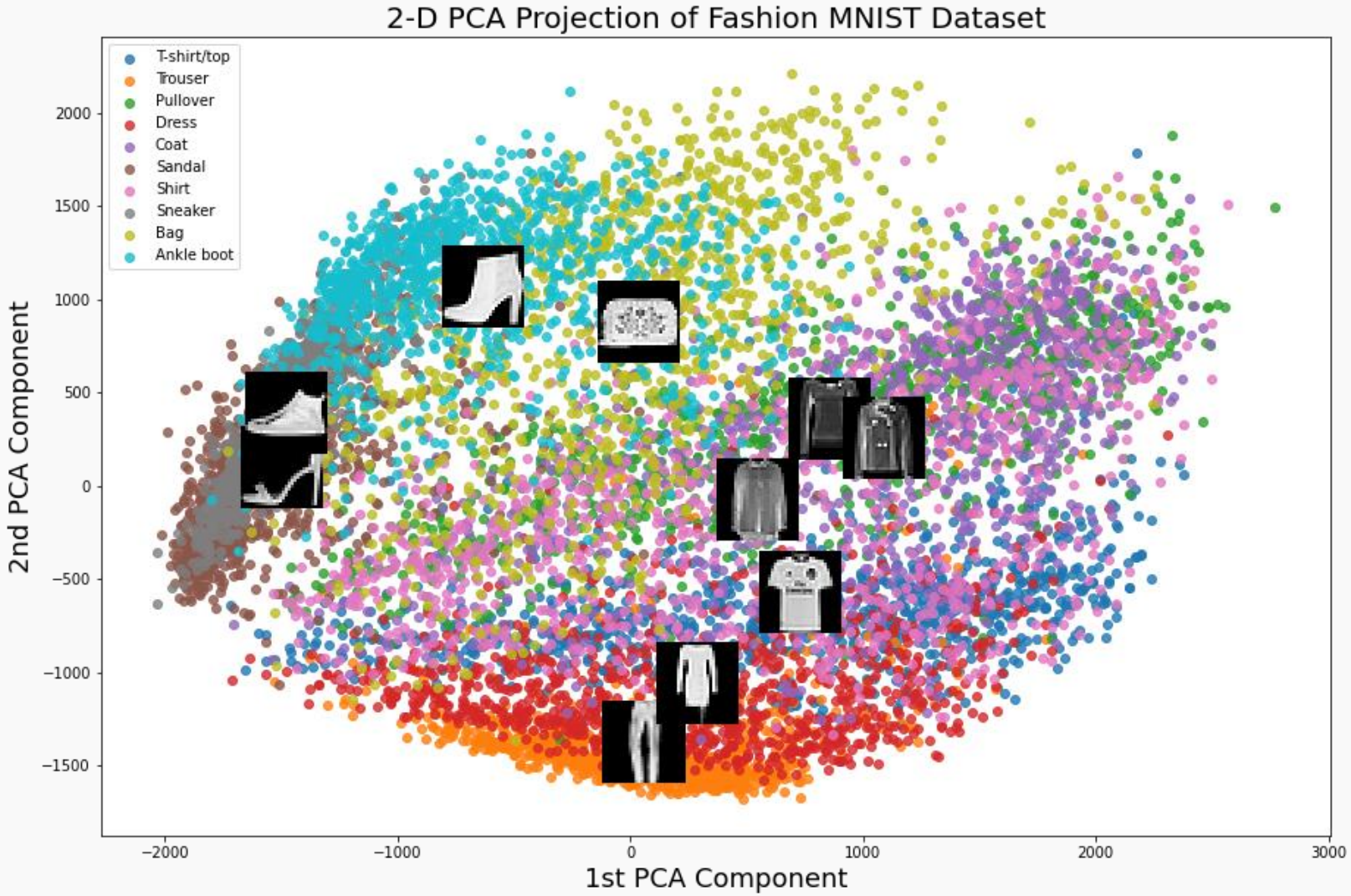
Explained Variance



3 Penguin Species: Interpretability



PCA for visualizing image data



Displaying examples at the center of their respective cluster we can appreciate that similar categories are close to one another in the projected space.

All this was done by finding linear combinations of pixels that explained the most variance in the data!

PCR

PCA for Regression (PCR)

PCA Step: First, can we use the components derived from PCA to transform the original predictors, X, into a set of new uncorrelated predictors, Z.

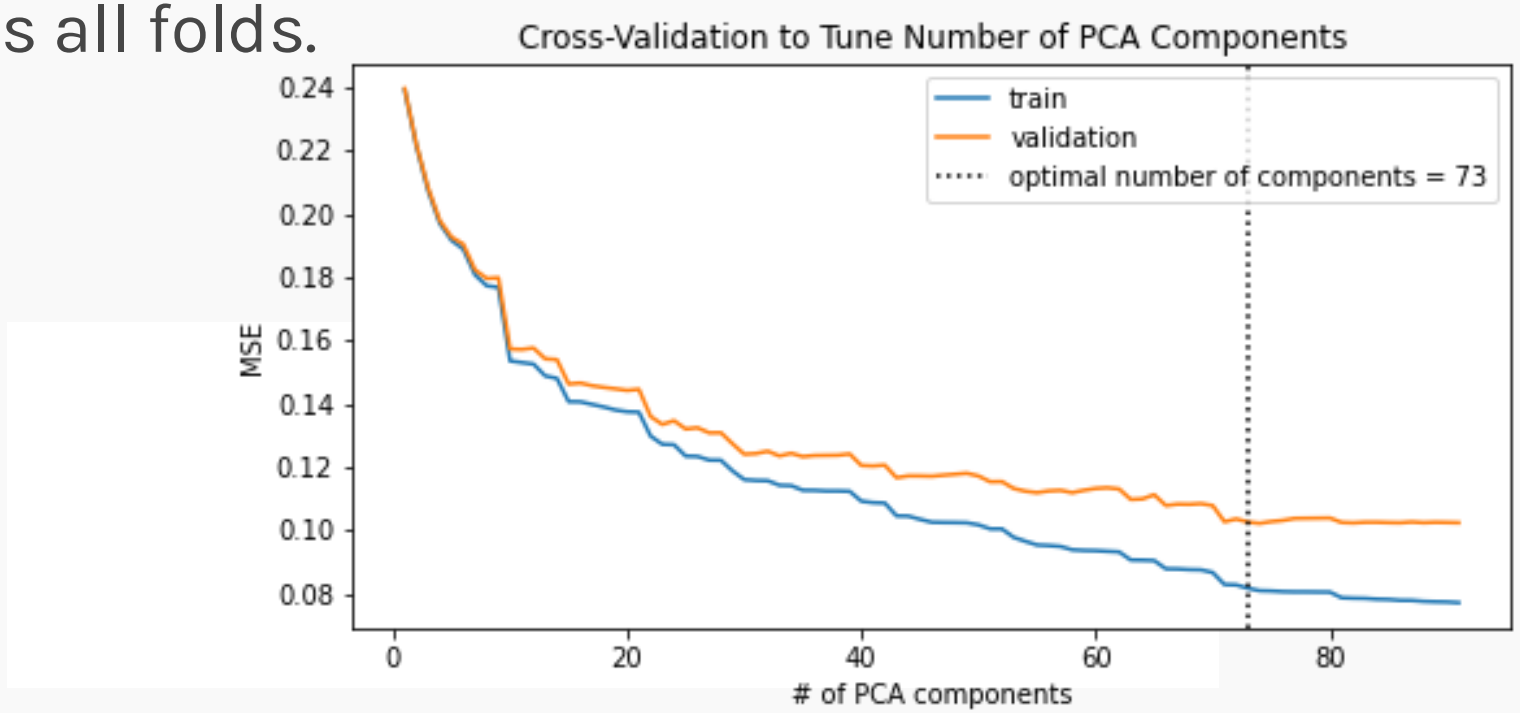
Regression Step: We then perform linear regression using a subset of these Z predictors.

By using a subset, PCR can simplify the model, reduce overfitting, and improve performance on unseen data.

Cross-validation to tune # of components in PCR

Looking only at variance explained **does not consider model performance!**

If modeling is your end goal, then you can use cross-validation to *tune* the number of components like any other hyperparameter, selecting the number that received the best mean validation error across all folds.



Missingness

Types of Missingness

There are 3 major types of missingness to be concerned about:

1. **Missing Completely at Random (MCAR)** - the probability of missingness in a variable is the same for all units. Like randomly poking holes in a data set.
2. **Missing at Random (MAR)** - the probability of missingness in a variable depends only on available information (in other predictors).
3. **Missing Not at Random (MNAR)** - the probability of missingness depends on information that has not been recorded and this information also predicts the missing values.

Missing completely at random (MCAR)

Missing Completely at Random is the best-case scenario, and the easiest to handle:

- Examples: a coin is flipped to determine whether an entry is removed. Or when values were just randomly missed when being entered in the computer.
- Effect if you ignore: there is no effect on inferences.
- How to handle: lots of options, but best to impute (more on next slide).

Missing at random (MAR)

Missing at Random is still a case that can be handled.

- Example(s): men and women respond at different rates to the question, "have you ever felt harassed at work?" (and may be harassed at different rates).
- Effect if you ignore: inferences are biased, and predictions usually suffer.
- How to handle: use the information in the other predictors to build a model and **impute** a value for the missing entry.

Key: we can fix any biases by modeling and imputing the missing values based on what is observed!

Missing Not at Random (MNAR)

Missing Not at Random is the worst-case scenario, and impossible to handle properly:

- Example(s): patients drop out of a study because they experience some bad side effect that was not measured. Or cheaters are less likely to respond when asked if you've ever cheated.
- Effect if you ignore: there are major effects on inferences or predictions.
- How to handle: you can 'improve' things by dealing with it like it is MAR, but you [likely] may never completely fix the bias. Simply incorporating a **missingness indicator variable** may be the best approach (if it a predictor that is missing).

Thank You