Machine Learning

Assessing and Improving ML Models

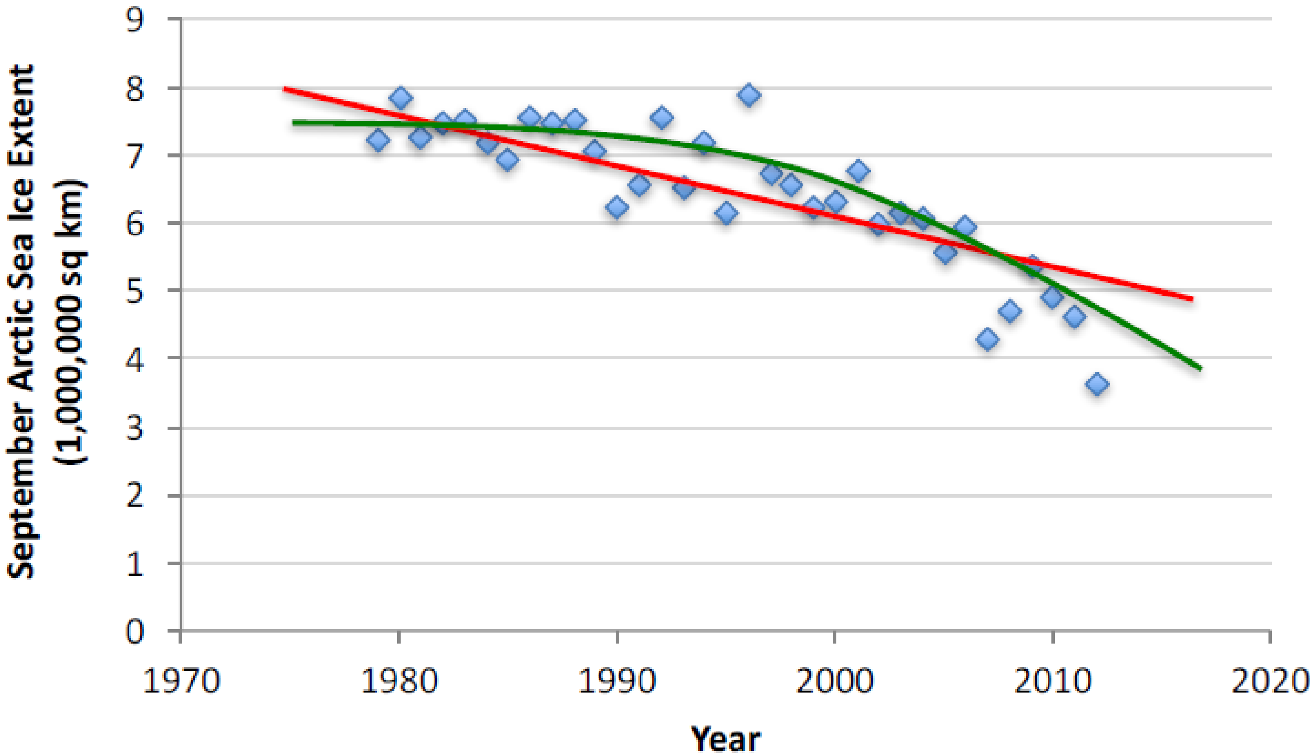
Indian Institute of Information Technology
Sri City, Chittoor

Today's Agenda

- Recap to regression problem
- How to evaluate the regression model?
- Evaluation Metrics for a regression model
- Dataset: Train, Validation and Test sets
- Train, test and validation split
- Data Sampling Methods
- Overfit and Underfit

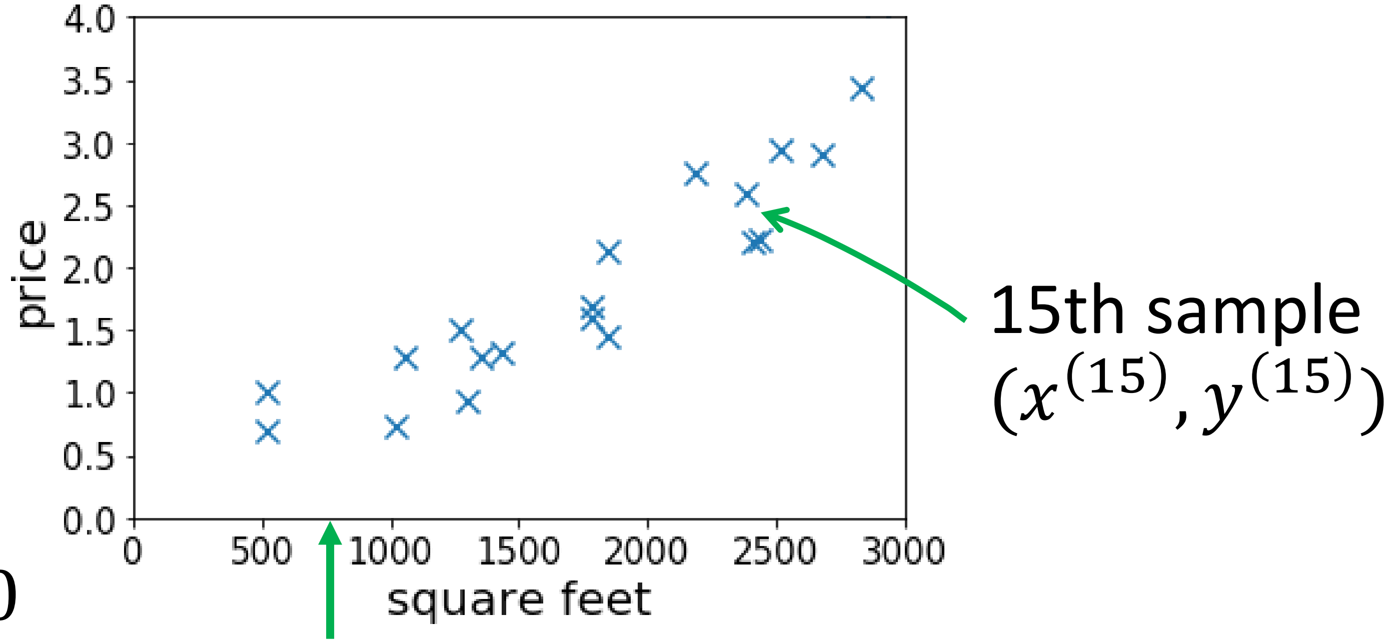
What is Regression Problem?

- Given $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$
- Learn a function $f(x)$ to predict y given x
  - y is real-valued == regression



Housing Price Prediction

- Given: a dataset that contains n samples
$$\left(x^{(1)}, y^{(1)}\right), \ldots \left(x^{(n)}, y^{(n)}\right)$$
- Task: If a residence has x square feet, predict its price?



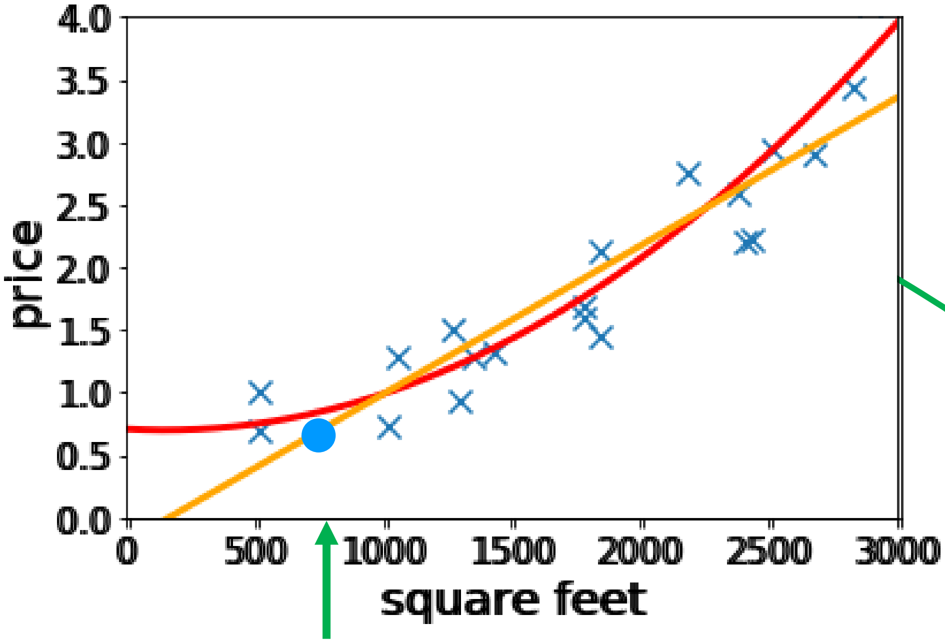
$x = 800$
$y = ?$

Housing Price Prediction

- Given: a dataset that contains n samples

$$\left(x^{(1)}, y^{(1)}\right), \ldots \left(x^{(n)}, y^{(n)}\right)$$

- Task: If a residence has x square feet, predict its price?



$x = 800$
$y = ?$

15th sample
$\left(x^{(15)}, y^{(15)}\right)$

- Solution: fitting linear/quadratic functions to the dataset.

How to evaluate the regression model?

- Cool, so what is the accuracy of your model prediction?
- Compared to classification (where you have discrete set of labels as prediction) predicting the accuracy in regression (where you have continuous real value number) is slightly difficult.!
- It might be impossible for your ML model to predict the exact value.
- So to calculate the accuracy, you can compared the predicted value as how close it is against the real value.

Evaluation Metrics for a Regression model

- *There are three main metrics used for evaluating the regression models:*

  - *R Square/Square of the Correlation Coefficient*

  - *Mean Square Error(MSE)/Root Mean Square Error(RMSE)*

  - *Mean Absolute Error(MAE)*

Evaluating a regression model: R square value

- *R Square/coefficient of determination:* It measures the proportion of the variance in the dependent variable that is predictable from the independent variable(s).

- *R* Square value is between 0 to 1 and bigger value indicates a better fit between prediction and actual value

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

Where y_i is the original value, $\hat{y}_i$ is the predicted value and $\bar{y}$ is the mean of original values.

Evaluating a regression model: Mean Squared Error (MSE)

- MSE is calculated by the sum of square of prediction error which is real output minus predicted output and then divide by the number of data points.
- It gives you an absolute number on how much your predicted results deviate from the actual number.
- Root Mean Square Error(RMSE) is the square root of MSE.

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

Where y_i is the original value and $\hat{y}_i$ is the predicted value.

Evaluating a regression model: Mean Absolute Error (MAE)

- Mean Absolute Error(MAE) is similar to Mean Square Error(MSE).
- Unlike MSE where we take the sum of square of errors, MAE computes the sum of absolute value of error.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|$$

Where y_i is the original value and $\hat{y}_i$ is the predicted value.

- Note: MSE gives larger penalisation to big prediction error by square it while MAE treats all errors the same.

Dataset: Train, Validation and Test sets

- **Train set:** The model is initially fit on a training dataset,[3] which is a set of examples used to fit the parameters (e.g. weights of connections between neurons in artificial neural networks) of the model.

- In practice, the training dataset often consists of pairs of an input vector (or scalar) and the corresponding output vector (or scalar), where the answer key is commonly denoted as the target (or label).

Dataset: Train, Validation and Test sets

- **Validation set:** Successively, the fitted model is used to predict the responses for the observations in a second dataset called the validation dataset.

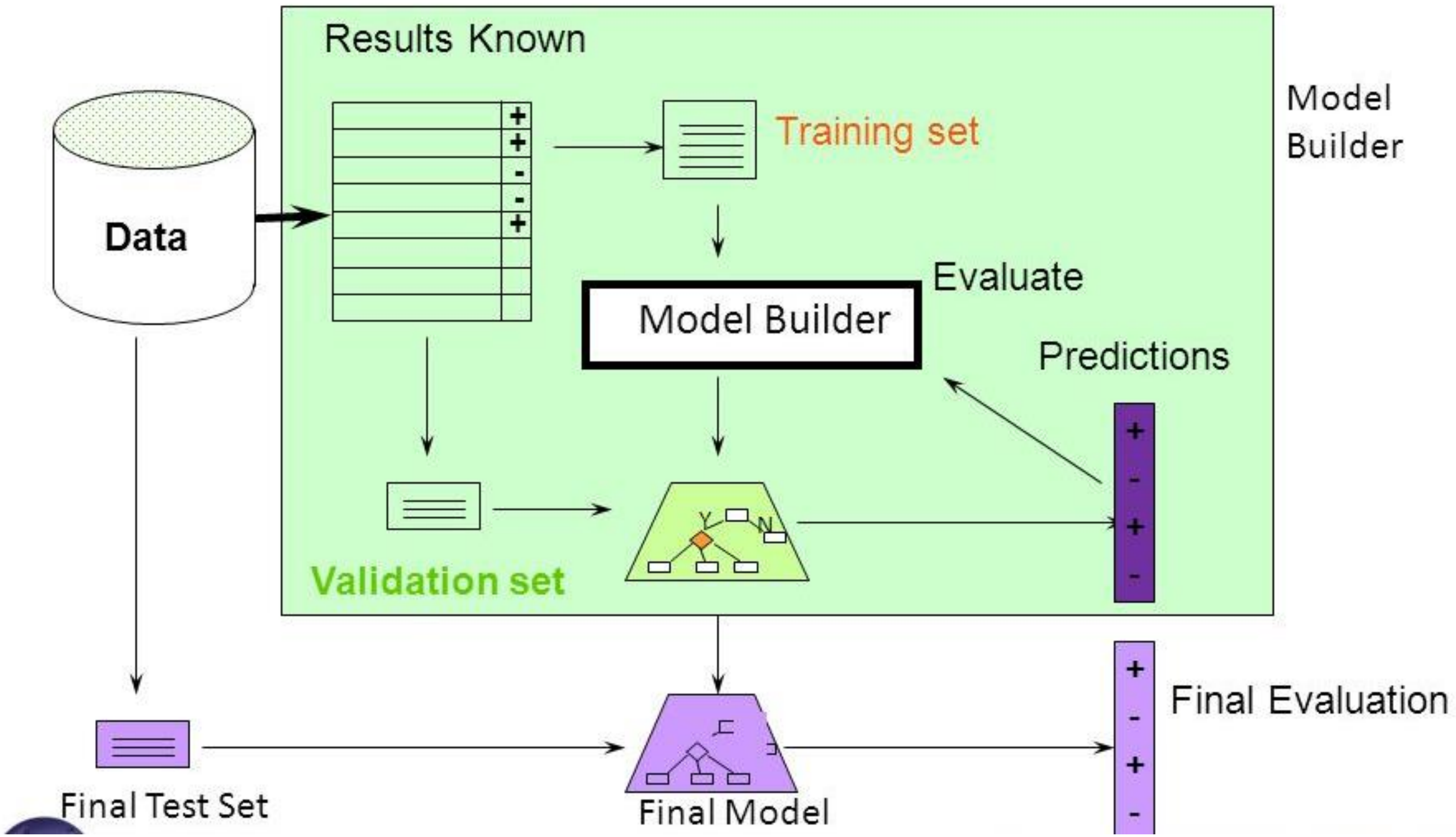
- The validation dataset provides an unbiased evaluation of a model fit on the training dataset while tuning the model's hyperparameters e.g. the number of hidden units (layers and layer widths) in a neural network.

- The validation dataset functions as a hybrid: it is training data used for testing, but neither as part of the low-level training nor as part of the final testing.

Dataset: Train, Validation and Test sets

- **Test set:** Finally, the test dataset is a dataset used to provide an unbiased evaluation of a final model fit on the training dataset.
- The test dataset is typically used to assess the final model that is selected during the validation process.
- It is only used once a model is completely trained(using the train and validation sets).
  - Many a times the validation set is used as the test set, but it is not good practice.
- The test set generally contains carefully sampled data that spans the various classes that the model would face, when used in the real world.

Train, test and validation split



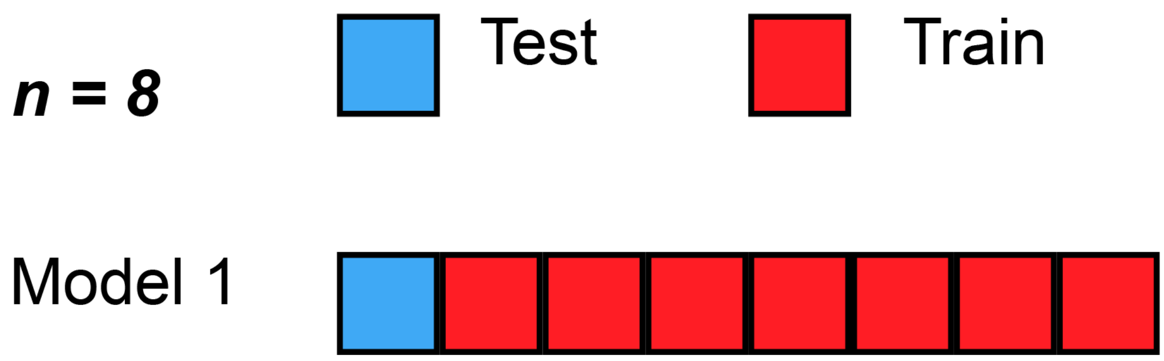
Data Sampling Methods

- **Hold out method**: In the holdout method, we randomly assign data points to two sets d0 and d1, usually called the training set and the test set, respectively.

- The size of each of the sets is arbitrary although typically the test set is smaller than the training set.

Data Sampling Methods

- **Cross Validation Method**: Cross-validation involves partitioning a sample of data into complementary subsets, performing the analysis on one subset (called the training set), and validating the analysis on the other subset (Validation set).
- To reduce variability, in most methods multiple rounds of cross-validation are performed using different partitions, and the validation results are combined (e.g. averaged) over the rounds to give an estimate of the model's predictive performance.



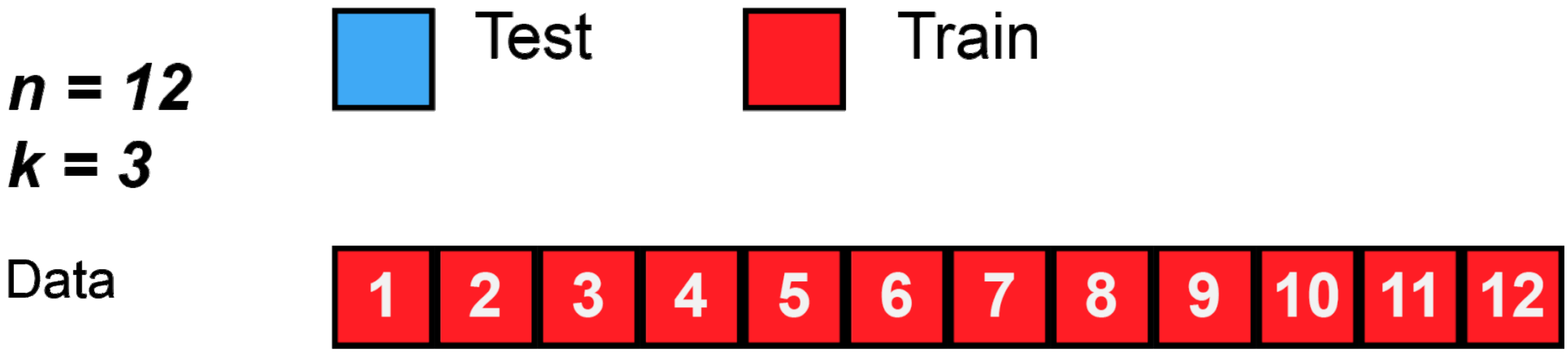
Data Sampling Methods

k-fold Cross validation: This procedure has a single parameter called “k” that refers to the number of groups that a given data sample is to be split into.

The general procedure is as follows:

1. Shuffle the dataset randomly.
2. Split the dataset into k groups
3. For each unique group:
    1. Take the group as a hold out or test data set
    2. Take the remaining groups as a training data set
    3. Fit a model on the training set and evaluate it on the test set
    4. Retain the evaluation score and discard the model
4. Summarize the skill of the model using the sample of model evaluation scores

k-fold Cross validation:



Overfit vs Underfit

- **Overfitting:** refers to a model that models the training data too well.

- Overfitting happens when a model learns the detail and noise in the training data to the extent that it negatively impacts the performance of the model on new data.

- In overfitting, ML model learns the inference rules based on the noise or random fluctuations in the training data.

Overfit vs Underfit

- **Underfitting:** refers to a model that can neither model the training data nor generalize to new data.

- An underfit machine learning model is not a suitable model and will be obvious as it will have poor performance on the training data.

- Underfitting is often not discussed as it is easy to detect given a good performance metric. The remedy is to move on and try alternate machine learning algorithms.

Thank You: Question?