Introduction to Machine Learning

Measures of Accuracy

Andres Mendez-Vazquez

August 18, 2025

Outline

1. Bias-Variance Dilemma
   - Introduction
   - Measuring the difference between optimal and learned
   - The Bias-Variance
   - "Extreme" Example
2. Confusion Matrix
   - Introduction
   - Statistical Testing
   - The α and β errors
   - The Initial Confusion Matrix
     - Metrics from the Confusion Matrix
   - The Multi Class Problem
   - Application in Computer Vision
   - Precision-Recall Curve
     - Application: Average Precision
   - Precision and Recall in the BERT Language Model
3. Receiver Operator Curves (ROC)
   - Introduction
   - Example
   - Algorithm for the ROC Curve
   - Area Under the Curve (AUC)
   - Other Measures: F_1-Measure
4. Cross Validation
   - Introduction
   - How to choose K
   - Types of Cross Validation
   - Solving The Imbalanced Class Problem

Outline

1. Bias-Variance Dilemma
   - Introduction
   - Measuring the difference between optimal and learned
   - The Bias-Variance
   - "Extreme" Example
2. Confusion Matrix
   - Introduction
   - Statistical Testing
   - The α and β errors
   - The Initial Confusion Matrix
     - Metrics from the Confusion Matrix
   - The Multi Class Problem
   - Application in Computer Vision
   - Precision-Recall Curve
     - Application: Average Precision
   - Precision and Recall in the BERT Language Model
3. Receiver Operator Curves (ROC)
   - Introduction
   - Example
   - Algorithm for the ROC Curve
   - Area Under the Curve (AUC)
   - Other Measures: F_1-Measure
4. Cross Validation
   - Introduction
   - How to choose K
   - Types of Cross Validation
   - Solving The Imbalanced Class Problem

Introduction

What did we see until now?

The design of learning machines from two main points:

- Statistical Point of View
- Linear Algebra and Optimization Point of View

Introduction

What did we see until now?

The design of learning machines from two main points:

- Statistical Point of View
- Linear Algebra and Optimization Point of View

Going back to the probability models

We might think in the machine to be learned as a function $g(\boldsymbol{x}|\mathcal{D})$....

- Something as curve fitting...

Introduction

What did we see until now?

The design of learning machines from two main points:

- Statistical Point of View
- Linear Algebra and Optimization Point of View

Going back to the probability models

We might think in the machine to be learned as a function $g(\boldsymbol{x}|\mathcal{D})$....

- Something as curve fitting...

Under a data set

$$\mathcal{D} = \{(\boldsymbol{x}_i, y_i) \,|i = 1, 2, ..., N\} \tag{1}$$

Remark: Where the $\boldsymbol{x}_i \sim p(\boldsymbol{x}|\Theta)$!!!

Thus, we have that

Two main functions

- A function $g(\boldsymbol{x}|\mathcal{D})$ obtained using some algorithm!!!
- For example, $E[y|\boldsymbol{x}]$ the optimal regression...

Thus, we have that

Two main functions

- A function $g(\boldsymbol{x}|\mathcal{D})$ obtained using some algorithm!!!
- For example, $E[y|\boldsymbol{x}]$ the optimal regression...

Important

The key factor here is the dependence of the approximation on $\mathcal{D}$.

Thus, we have that

Two main functions

- A function $g(\boldsymbol{x}|\mathcal{D})$ obtained using some algorithm!!!
- For example, $E[y|\boldsymbol{x}]$ the optimal regression...

Important

The key factor here is the dependence of the approximation on $\mathcal{D}$.

Why?

The approximation may be very good for a specific training data set but very bad for another.

- This is the reason of studying fusion of information at decision level...

Outline

1. Bias-Variance Dilemma
   - Introduction
   - Measuring the difference between optimal and learned
   - The Bias-Variance
   - "Extreme" Example
2. Confusion Matrix
   - Introduction
   - Statistical Testing
   - The α and β errors
   - The Initial Confusion Matrix
     - Metrics from the Confusion Matrix
   - The Multi Class Problem
   - Application in Computer Vision
   - Precision-Recall Curve
     - Application: Average Precision
   - Precision and Recall in the BERT Language Model
3. Receiver Operator Curves (ROC)
   - Introduction
   - Example
   - Algorithm for the ROC Curve
   - Area Under the Curve (AUC)
   - Other Measures: F_1-Measure
4. Cross Validation
   - Introduction
   - How to choose K
   - Types of Cross Validation
   - Solving The Imbalanced Class Problem

How do we measure the difference?

We can use the variance

$$Var(X) = E((X - \mu)^2)$$

How do we measure the difference?

We can use the variance

$$Var(X) = E((X - \mu)^2)$$

We can do that for our data

$$Var_{\mathcal{D}}\left(g\left(\boldsymbol{x}|\mathcal{D}\right)\right) = E_D\left(\left(g\left(\boldsymbol{x}|\mathcal{D}\right) - E\left[y|\boldsymbol{x}\right]\right)^2\right)$$

How do we measure the difference?

We can use the variance

$$Var(X) = E((X - \mu)^2)$$

We can do that for our data

$$Var_{\mathcal{D}}\left(g\left(\boldsymbol{x}|\mathcal{D}\right)\right) = E_D\left(\left(g\left(\boldsymbol{x}|\mathcal{D}\right) - E\left[y|\boldsymbol{x}\right]\right)^2\right)$$

Now, if we add and subtract

$$E_D\left[g\left(\boldsymbol{x}|\mathcal{D}\right)\right] \tag{2}$$

How do we measure the difference?

We can use the variance

$$Var(X) = E((X - \mu)^2)$$

We can do that for our data

$$Var_{\mathcal{D}}\left(g\left(\boldsymbol{x}|\mathcal{D}\right)\right) = E_D\left(\left(g\left(\boldsymbol{x}|\mathcal{D}\right) - E\left[y|\boldsymbol{x}\right]\right)^2\right)$$

Now, if we add and subtract

$$E_D\left[g\left(\boldsymbol{x}|\mathcal{D}\right)\right] \tag{2}$$

Remark: The expected output of the machine $g\left(\boldsymbol{x}|\mathcal{D}\right)$

Thus, we have that

Or Original variance

$$
\begin{aligned}
Var_{\mathcal{D}}\left(g\left(\boldsymbol{x}|\mathcal{D}\right)\right) &= E_D\left(\left(g\left(\boldsymbol{x}|\mathcal{D}\right) - E\left[y|\boldsymbol{x}\right]\right)^2\right) \\
&= E_D\left(\left(g\left(\boldsymbol{x}|\mathcal{D}\right) - E_D\left[g\left(\boldsymbol{x}|\mathcal{D}\right)\right] + E_D\left[g\left(\boldsymbol{x}|\mathcal{D}\right)\right] - E\left[y|\boldsymbol{x}\right]\right)^2\right) \\
&= E_D\left(\left(g\left(\boldsymbol{x}|\mathcal{D}\right) - E_D\left[g\left(\boldsymbol{x}|\mathcal{D}\right)\right]\right)^2 + ... \right. \\
&...2\left(\left(g\left(\boldsymbol{x}|\mathcal{D}\right) - E_D\left[g\left(\boldsymbol{x}|\mathcal{D}\right)\right]\right)\right)\left(E_D\left[g\left(\boldsymbol{x}|\mathcal{D}\right)\right] - E\left[y|\boldsymbol{x}\right]\right) + ... \\
&... \left.\left(E_D\left[g\left(\boldsymbol{x}|\mathcal{D}\right)\right] - E\left[y|\boldsymbol{x}\right]\right)^2\right)
\end{aligned}
$$

Thus, we have that

Or Original variance

$$\begin{aligned} Var_{\mathcal{D}}\left(g\left(\boldsymbol{x}|\mathcal{D}\right)\right) &= E_D\left(\left(g\left(\boldsymbol{x}|\mathcal{D}\right) - E\left[y|\boldsymbol{x}\right]\right)^2\right) \\ &= E_D\left(\left(g\left(\boldsymbol{x}|\mathcal{D}\right) - E_D\left[g\left(\boldsymbol{x}|\mathcal{D}\right)\right] + E_D\left[g\left(\boldsymbol{x}|\mathcal{D}\right)\right] - E\left[y|\boldsymbol{x}\right]\right)^2\right) \\ &= E_D\left(\left(g\left(\boldsymbol{x}|\mathcal{D}\right) - E_D\left[g\left(\boldsymbol{x}|\mathcal{D}\right)\right]\right)^2 + ... \\ &...2\left(\left(g\left(\boldsymbol{x}|\mathcal{D}\right) - E_D\left[g\left(\boldsymbol{x}|\mathcal{D}\right)\right]\right)\right)\left(E_D\left[g\left(\boldsymbol{x}|\mathcal{D}\right)\right] - E\left[y|\boldsymbol{x}\right]\right) + ... \\ &... \left(E_D\left[g\left(\boldsymbol{x}|\mathcal{D}\right)\right] - E\left[y|\boldsymbol{x}\right]\right)^2\Big) \end{aligned}$$

Finally

$$E_D\left(\left(\left(g\left(\boldsymbol{x}|\mathcal{D}\right) - E_D\left[g\left(\boldsymbol{x}|\mathcal{D}\right)\right]\right)\right)\left(E_D\left[g\left(\boldsymbol{x}|\mathcal{D}\right)\right] - E\left[y|\boldsymbol{x}\right]\right)\right) =? \quad (3)$$

Outline

1. Bias-Variance Dilemma
   - Introduction
   - Measuring the difference between optimal and learned
   - The Bias-Variance
   - "Extreme" Example
2. Confusion Matrix
   - Introduction
   - Statistical Testing
   - The α and β errors
   - The Initial Confusion Matrix
     - Metrics from the Confusion Matrix
   - The Multi Class Problem
   - Application in Computer Vision
   - Precision-Recall Curve
     - Application: Average Precision
   - Precision and Recall in the BERT Language Model
3. Receiver Operator Curves (ROC)
   - Introduction
   - Example
   - Algorithm for the ROC Curve
   - Area Under the Curve (AUC)
   - Other Measures: F_1-Measure
4. Cross Validation
   - Introduction
   - How to choose K
   - Types of Cross Validation
   - Solving The Imbalanced Class Problem

We have the Bias-Variance

Our Final Equation

$$E_D\left(\left(g\left(\boldsymbol{x}|\mathcal{D}\right)-E\left[y|\boldsymbol{x}\right]\right)^2\right)=\underbrace{E_D\left(\left(g\left(\boldsymbol{x}|\mathcal{D}\right)-E_D\left[g\left(\boldsymbol{x}|\mathcal{D}\right)\right]\right)^2\right)}_{VARIANCE}+\underbrace{\left(E_D\left[g\left(\boldsymbol{x}|\mathcal{D}\right)\right]-E\left[y|\boldsymbol{x}\right]\right)^2}_{BIAS}$$

We have the Bias-Variance

Our Final Equation

$$E_D\left(\left(g\left(\boldsymbol{x}|\mathcal{D}\right)-E\left[y|\boldsymbol{x}\right]\right)^2\right)=\underbrace{E_D\left(\left(g\left(\boldsymbol{x}|\mathcal{D}\right)-E_D\left[g\left(\boldsymbol{x}|\mathcal{D}\right)\right]\right)^2\right)}_{VARIANCE}+\underbrace{\left(E_D\left[g\left(\boldsymbol{x}|\mathcal{D}\right)\right]-E\left[y|\boldsymbol{x}\right]\right)^2}_{BIAS}$$

Where the variance

It represents the measure of the error between our machine $g\left(\boldsymbol{x}|\mathcal{D}\right)$ and the expected output of the machine under $\boldsymbol{x}_i \sim p\left(\boldsymbol{x}|\Theta\right)$.

We have the Bias-Variance

Our Final Equation

$$E_D\left(\left(g\left(\boldsymbol{x}|\mathcal{D}\right)-E\left[y|\boldsymbol{x}\right]\right)^2\right)=\underbrace{E_D\left(\left(g\left(\boldsymbol{x}|\mathcal{D}\right)-E_D\left[g\left(\boldsymbol{x}|\mathcal{D}\right)\right]\right)^2\right)}_{VARIANCE}+\underbrace{\left(E_D\left[g\left(\boldsymbol{x}|\mathcal{D}\right)\right]-E\left[y|\boldsymbol{x}\right]\right)^2}_{BIAS}$$

Where the variance

It represents the measure of the error between our machine $g\left(\boldsymbol{x}|\mathcal{D}\right)$ and the expected output of the machine under $\boldsymbol{x}_i\sim p\left(\boldsymbol{x}|\Theta\right)$.

Where the bias

It represents the quadratic error between the expected output of the machine under $\boldsymbol{x}_i\sim p\left(\boldsymbol{x}|\Theta\right)$ and the expected output of the optimal regression.

Remarks

We have then

Even if the estimator is unbiased, it can still result in a large mean square error due to a large variance term.

Remarks

We have then

Even if the estimator is unbiased, it can still result in a large mean square error due to a large variance term.

The situation is more dire in a finite set of data $\mathcal{D}$

We have then a trade-off:

1. Increasing the bias decreases the variance and vice versa.

Remarks

We have then

Even if the estimator is unbiased, it can still result in a large mean square error due to a large variance term.

The situation is more dire in a finite set of data $\mathcal{D}$

We have then a trade-off:

1. Increasing the bias decreases the variance and vice versa.
2. This is known as the **bias–variance dilemma**.

Similar to...

Curve Fitting

If, for example, the adopted model is complex (many parameters involved) with respect to the number N, the model will fit the idiosyncrasies of the specific data set.

Similar to...

Curve Fitting

If, for example, the adopted model is complex (many parameters involved) with respect to the number N, the model will fit the idiosyncrasies of the specific data set.

Thus

Thus, it will result in low bias but will yield high variance, as we change from one data set to another data set.

Similar to...

Curve Fitting

If, for example, the adopted model is complex (many parameters involved) with respect to the number N, the model will fit the idiosyncrasies of the specific data set.

Thus

Thus, it will result in low bias but will yield high variance, as we change from one data set to another data set.

Furthermore

If N grows we can have a more complex model to be fitted which reduces bias and ensures low variance.

Similar to...

Curve Fitting

If, for example, the adopted model is complex (many parameters involved) with respect to the number N, the model will fit the idiosyncrasies of the specific data set.

Thus

Thus, it will result in low bias but will yield high variance, as we change from one data set to another data set.

Furthermore

If N grows we can have a more complex model to be fitted which reduces bias and ensures low variance.

- However, N is always finite!!!

Thus

You always need to compromise

However, you always have some a priori knowledge about the data

Thus

You always need to compromise

However, you always have some a priori knowledge about the data

Allowing you to impose restrictions

Lowering the bias and the variance

Thus

You always need to compromise

However, you always have some a priori knowledge about the data

Allowing you to impose restrictions

Lowering the bias and the variance

Nevertheless

We have the following example to grasp better the bothersome **bias–variance dilemma**.

For this

Assume

The data is generated by the following function

$$y = f(x) + \epsilon,$$

$$\epsilon \sim \mathcal{N}\left(0, \sigma_\epsilon^2\right)$$

For this

Assume

The data is generated by the following function

$$y = f(x) + \epsilon,$$
$$\epsilon \sim \mathcal{N}\left(0, \sigma_\epsilon^2\right)$$

We know that

The optimum regressor is $E[y|x] = f(x)$

For this

Assume

The data is generated by the following function

$$y = f(x) + \epsilon,$$
$$\epsilon \sim \mathcal{N}\left(0, \sigma_\epsilon^2\right)$$

We know that

The optimum regressor is $E[y|x] = f(x)$

Furthermore

Assume that the randomness in the different training sets, $\mathcal{D}$, is due to the y_i's (Affected by noise), while the respective points, x_i, are fixed.

Outline

1. Bias-Variance Dilemma
   - Introduction
   - Measuring the difference between optimal and learned
   - The Bias-Variance
   - "Extreme" Example
2. Confusion Matrix
   - Introduction
   - Statistical Testing
   - The α and β errors
   - The Initial Confusion Matrix
     - Metrics from the Confusion Matrix
   - The Multi Class Problem
   - Application in Computer Vision
   - Precision-Recall Curve
     - Application: Average Precision
   - Precision and Recall in the BERT Language Model
3. Receiver Operator Curves (ROC)
   - Introduction
   - Example
   - Algorithm for the ROC Curve
   - Area Under the Curve (AUC)
   - Other Measures: F_1-Measure
4. Cross Validation
   - Introduction
   - How to choose K
   - Types of Cross Validation
   - Solving The Imbalanced Class Problem

Sampling the Space

Imagine that $\mathcal{D} \subset [x_1, x_2]$ in which x lies

For example, you can choose $x_i = x_1 + \frac{x_2 - x_1}{N-1}(i-1)$ with $i = 1, 2, ..., N$

Case 1

Choose the estimate of $f(x)$, $g(x|\mathcal{D})$, to be independent of $\mathcal{D}$

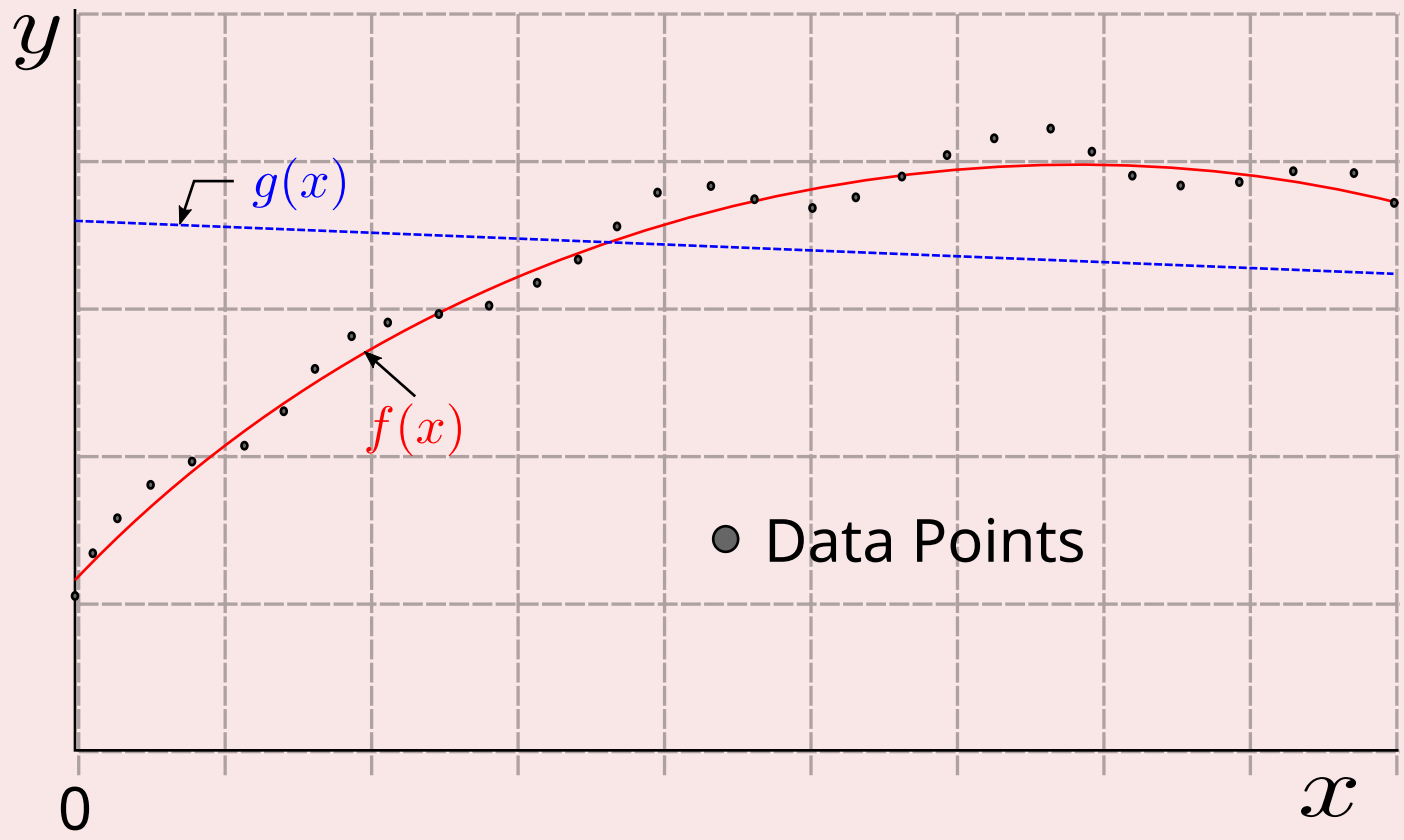
For example, $g(x) = w_1 x + w_0$

Case 1

Choose the estimate of $f(x)$, $g(x|\mathcal{D})$, to be independent of $\mathcal{D}$

For example, $g(x) = w_1 x + w_0$

For example, the points are spread around $(x, f(x))$



Case 1

Since $g(x)$ is fixed

$$E_{\mathcal{D}}\left[g\left(x|\mathcal{D}\right)\right] = g\left(x|\mathcal{D}\right) \equiv g\left(x\right) \tag{4}$$

Case 1

Since $g(x)$ is fixed

$$E_{\mathcal{D}}\left[g\left(x|\mathcal{D}\right)\right] = g\left(x|\mathcal{D}\right) \equiv g\left(x\right) \tag{4}$$

With

$$Var_{\mathcal{D}}\left[g\left(x|\mathcal{D}\right)\right] = 0 \tag{5}$$

Case 1

Since $g(x)$ is fixed

$$E_{\mathcal{D}}\left[g\left(x|\mathcal{D}\right)\right] = g\left(x|\mathcal{D}\right) \equiv g\left(x\right) \tag{4}$$

With

$$Var_{\mathcal{D}}\left[g\left(x|\mathcal{D}\right)\right] = 0 \tag{5}$$

On the other hand

Because $g(x)$ was chosen arbitrarily the expected bias must be large.

$$\underbrace{\left(E_D\left[g\left(\boldsymbol{x}|\mathcal{D}\right)\right] - E\left[y|\boldsymbol{x}\right]\right)^2}_{BIAS} \tag{6}$$

Case 2

In the other hand

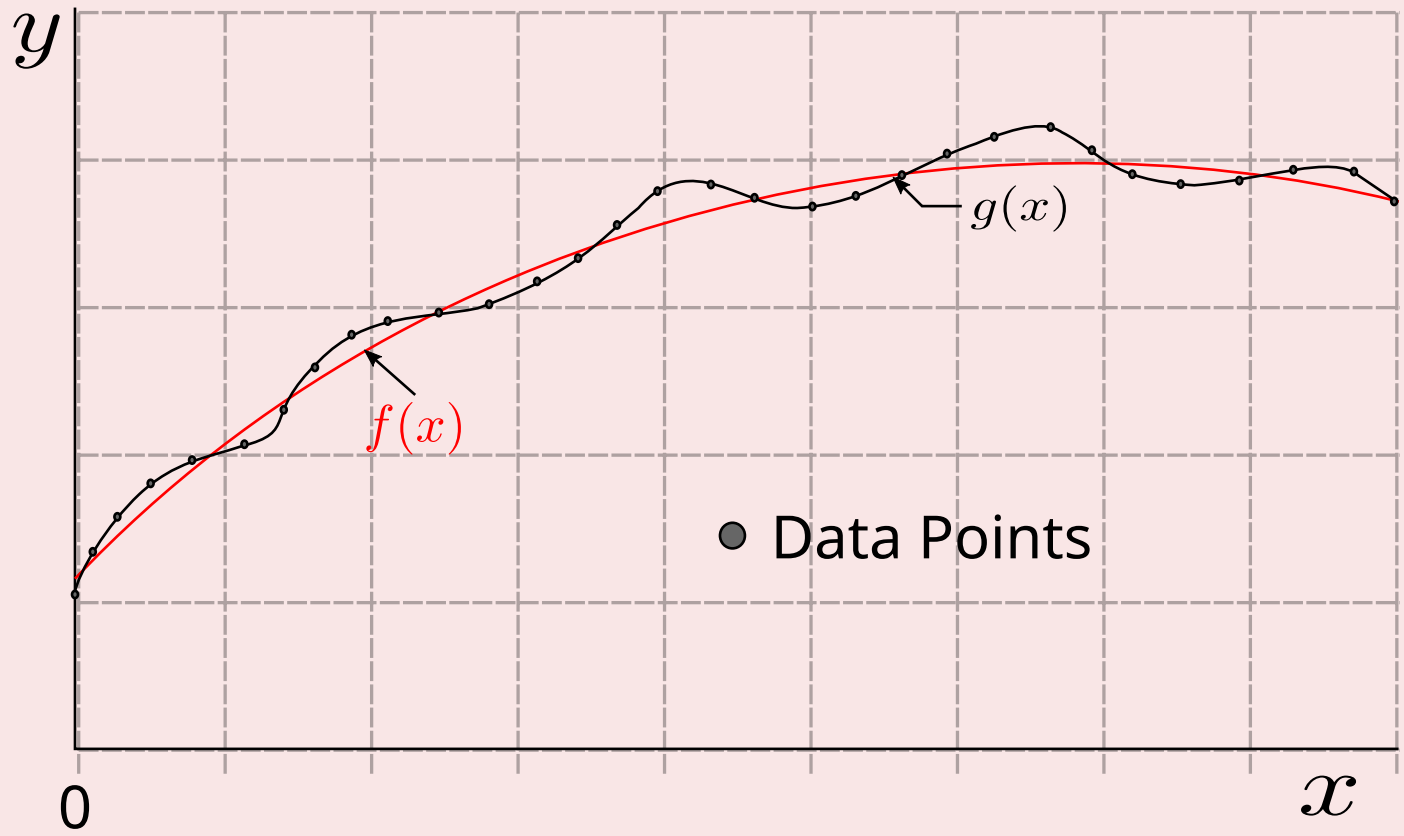
Now, $g_1(x)$ corresponds to a polynomial of high degree so it can pass through each training point in $\mathcal{D}$.

Case 2

In the other hand

Now, $g_1(x)$ corresponds to a polynomial of high degree so it can pass through each training point in $\mathcal{D}$.

Example of $g_1(x)$



Case 2

Due to the zero mean of the noise source

$$E_D\left[g_1\left(\boldsymbol{x}|\mathcal{D}\right)\right] = f\left(x\right) = E\left[y|x\right] \text{ for any } x = x_i \tag{7}$$

Remark: At the training points the bias is zero.

Case 2

Due to the zero mean of the noise source

$$E_D\left[g_1\left(\boldsymbol{x}|\mathcal{D}\right)\right] = f\left(x\right) = E\left[y|x\right] \text{ for any } x = x_i \tag{7}$$

Remark: At the training points the bias is zero.

However the variance increases

$$\begin{aligned} E_D\left[\left(g_1\left(\boldsymbol{x}|\mathcal{D}\right) - E_D\left[g_1\left(\boldsymbol{x}|\mathcal{D}\right)\right]\right)^2\right] &= E_D\left[\left(f\left(x\right) + \epsilon - f\left(x\right)\right)^2\right] \\ &= \sigma_\epsilon^2, \text{ for } x = x_i, i = 1, 2, ..., N \end{aligned}$$

Case 2

Due to the zero mean of the noise source

$$E_D\left[g_1\left(\boldsymbol{x}|\mathcal{D}\right)\right] = f\left(x\right) = E\left[y|x\right] \text{ for any } x = x_i \tag{7}$$

Remark: At the training points the bias is zero.

However the variance increases

$$\begin{aligned} E_D\left[\left(g_1\left(\boldsymbol{x}|\mathcal{D}\right) - E_D\left[g_1\left(\boldsymbol{x}|\mathcal{D}\right)\right]\right)^2\right] &= E_D\left[\left(f\left(x\right) + \epsilon - f\left(x\right)\right)^2\right] \\ &= \sigma_\epsilon^2, \text{ for } x = x_i, i = 1, 2, ..., N \end{aligned}$$

In other words

The bias becomes zero (or approximately zero) but the variance is now equal to the variance of the noise source.

Observations

First

Everything that has been said so far applies to both the regression and the classification tasks.

Observations

First

Everything that has been said so far applies to both the regression and the classification tasks.

However

Mean squared error is not the best way to measure the power of a classifier.

Observations

First

Everything that has been said so far applies to both the regression and the classification tasks.

However

Mean squared error is not the best way to measure the power of a classifier.

Think about

A classifier that sends everything far away of the hyperplane!!! Away from the values $+-1$!!!

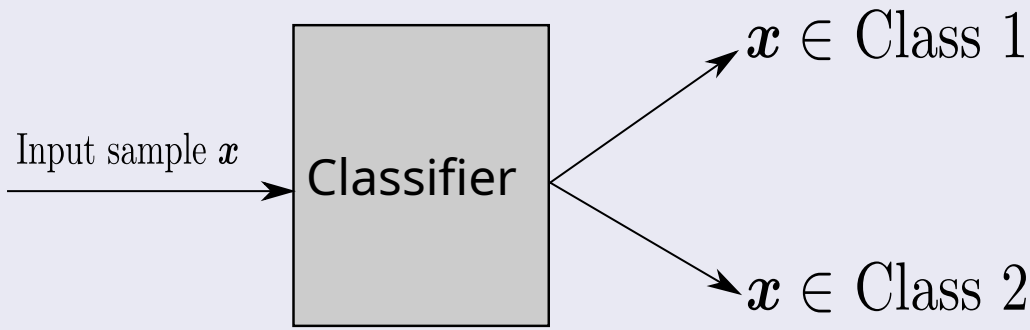
Outline

1. Bias-Variance Dilemma
   - Introduction
   - Measuring the difference between optimal and learned
   - The Bias-Variance
   - "Extreme" Example
2. Confusion Matrix
   - Introduction
   - Statistical Testing
   - The α and β errors
   - The Initial Confusion Matrix
     - Metrics from the Confusion Matrix
   - The Multi Class Problem
   - Application in Computer Vision
   - Precision-Recall Curve
     - Application: Average Precision
   - Precision and Recall in the BERT Language Model
3. Receiver Operator Curves (ROC)
   - Introduction
   - Example
   - Algorithm for the ROC Curve
   - Area Under the Curve (AUC)
   - Other Measures: F_1-Measure
4. Cross Validation
   - Introduction
   - How to choose K
   - Types of Cross Validation
   - Solving The Imbalanced Class Problem

Sooner of Latter you need to know how efficient is your algorithm

Thus, we need a measures of accuracy

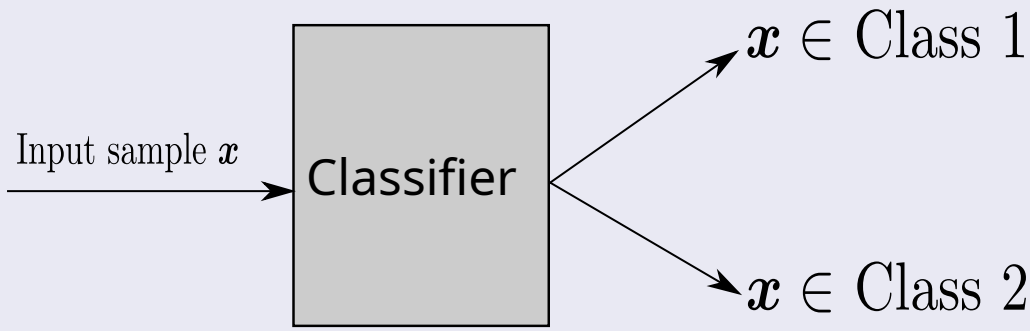
Thus, we begin with the classic classifier for two classes



Sooner of Latter you need to know how efficient is your algorithm

Thus, we need a measures of accuracy

Thus, we begin with the classic classifier for two classes



Here

A dataset used for performance evaluation is called a **test dataset.**

Therefore

It is a good idea to build a measure of performance

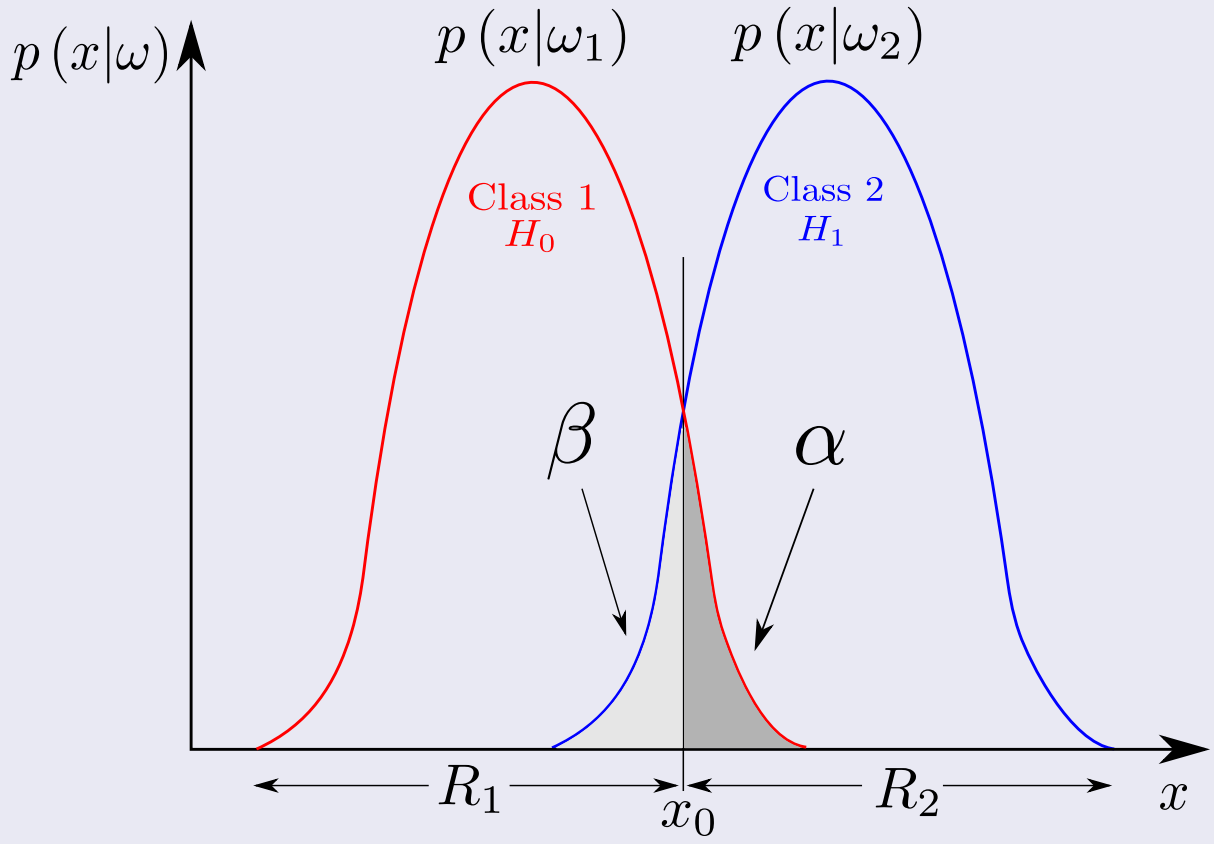
For this, we can use the idea of error in statistics.

Therefore

It is a good idea to build a measure of performance

For this, we can use the idea of error in statistics.

We have two distribution for each class



Outline

1. Bias-Variance Dilemma
   - Introduction
   - Measuring the difference between optimal and learned
   - The Bias-Variance
   - "Extreme" Example
2. Confusion Matrix
   - Introduction
   - Statistical Testing
   - The α and β errors
   - The Initial Confusion Matrix
     - Metrics from the Confusion Matrix
   - The Multi Class Problem
   - Application in Computer Vision
   - Precision-Recall Curve
     - Application: Average Precision
   - Precision and Recall in the BERT Language Model
3. Receiver Operator Curves (ROC)
   - Introduction
   - Example
   - Algorithm for the ROC Curve
   - Area Under the Curve (AUC)
   - Other Measures: F_1-Measure
4. Cross Validation
   - Introduction
   - How to choose K
   - Types of Cross Validation
   - Solving The Imbalanced Class Problem

Taking us to Statistical testing

From

- Neyman Jerzy and Pearson Egon Sharpe 1933IX. On the problem of the most efficient tests of statistical hypothesesPhilosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character231:289–337

Taking us to Statistical testing

From

- Neyman Jerzy and Pearson Egon Sharpe 1933IX. On the problem of the most efficient tests of statistical hypothesesPhilosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character231:289–337

Here we have two Hypothesis H_0 and H_1

- Each corresponding a single density probability function $p(x|\theta_0)$ and $p(x|\theta_1)$ such that

$$P(W|\theta_i) = \int_W p(x|\theta_i)\, dx = 1$$

Taking us to Statistical testing

From

- Neyman Jerzy and Pearson Egon Sharpe 1933IX. On the problem of the most efficient tests of statistical hypothesesPhilosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character231:289–337

Here we have two Hypothesis H_0 and H_1

- Each corresponding a single density probability function $p(x|\theta_0)$ and $p(x|\theta_1)$ such that

$$P(W|\theta_i) = \int_W p(x|\theta_i)\, dx = 1$$

Assume a sample point x in the W in the sample space

- Now consider all possible hypothesis based in that sample point $H_x = \{p(x|\theta)\}$

Then, the hypothesis H_0

Using the upper bound hypothesis $p(x|\theta_\omega)$

- We have that $\lambda = \frac{p(x|\theta_\omega)}{p(x|\theta_0)}$
- Which can be seen as $\lambda p(x|\theta_0) = p(x|\theta_\omega)$

Then, the hypothesis H_0

Using the upper bound hypothesis $p(x|\theta_\omega)$

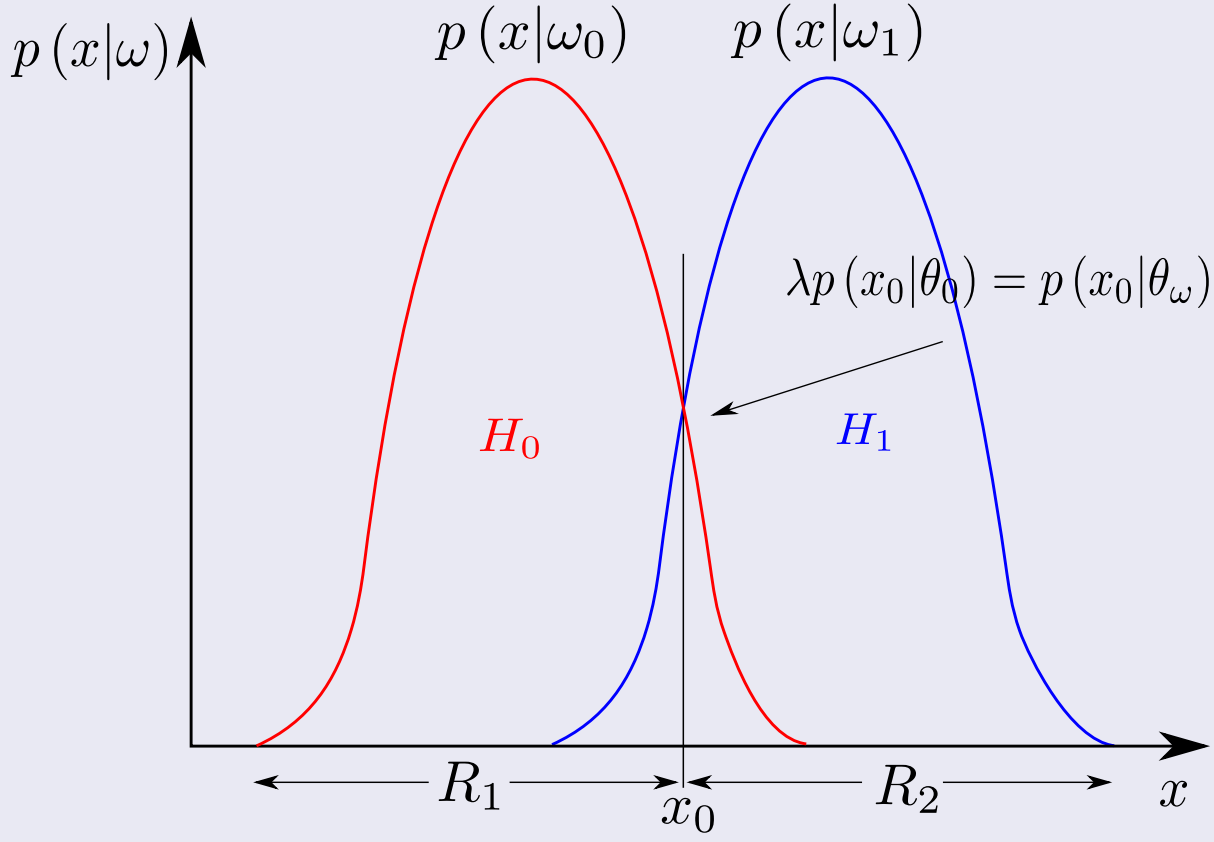
- We have that $\lambda = \frac{p(x|\theta_\omega)}{p(x|\theta_0)}$
- Which can be seen as $\lambda p(x|\theta_0) = p(x|\theta_\omega)$

Here we can ask the following

- What is the area where
  - $p(x|\theta_\omega) < \lambda p(x|\theta_0)$
  - $p(x|\theta_\omega) > \lambda p(x|\theta_\omega)$

What?

Remember



Actually, we are talking of rejection areas

Yes, where finding $x \in R_2$

- Rejection area, actually, thus we are asking to find $\lambda \geq 0$ (1 in the previous example) such that if
  - $x \in R \Longrightarrow p(x|\theta_\omega) > \lambda p(x|\theta_0)$
  - $x \in R^c \Longrightarrow p(x|\theta_\omega) < \lambda p(x|\theta_0)$

Neyman–Pearson Lemma

Consider a test with hypotheses $H_0 : \theta = \theta_0$ and $H_1 : \theta = \theta_1$

- where the probability density function (or probability mass function) is $p(x \mid \theta_i)$ for $i = 0, 1$.

Neyman–Pearson Lemma

Consider a test with hypotheses $H_0 : \theta = \theta_0$ and $H_1 : \theta = \theta_1$

- where the probability density function (or probability mass function) is $p(x \mid \theta_i)$ for $i = 0, 1$.

For any hypothesis test with rejection set R, and any $\alpha \in [0, 1]$

- we say that it satisfies condition P_α if $\alpha = P_{\theta_0}(X \in R)$

Neyman–Pearson Lemma

Consider a test with hypotheses $H_0 : \theta = \theta_0$ and $H_1 : \theta = \theta_1$

- where the probability density function (or probability mass function) is $p(x \mid \theta_i)$ for $i = 0, 1$.

For any hypothesis test with rejection set R, and any $\alpha \in [0, 1]$

- we say that it satisfies condition P_α if $\alpha = P_{\theta_0}(X \in R)$

That is, the test has size α

- That is, the probability of falsely rejecting the null hypothesis is α.

Thus, we have

$\exists \lambda \geq 0$ such that

- $x \in R - W \Longrightarrow p\left(x|\theta_\omega\right) > \lambda p\left(x|\theta_0\right)$
- $x \in R^c - W \Longrightarrow p\left(x|\theta_\omega\right) < \lambda p\left(x|\theta_0\right)$

Thus, we have

$\exists \lambda \geq 0$ such that

- $x \in R - W \Longrightarrow p\left(x|\theta_\omega\right) > \lambda p\left(x|\theta_0\right)$
- $x \in R^c - W \Longrightarrow p\left(x|\theta_\omega\right) < \lambda p\left(x|\theta_0\right)$

Where W is a negligible set in both θ_0 and θ_1 (Or Probability/Measure zero)

- $P_{\theta_0}\left(X \in R\right) = P_{\theta_1}\left(X \in R\right) = 0$
  - That is, we have a strict likelihood ratio test, except on a negligible subset.

Example

Let $x_1, ..., x_n$ be a random variables from $N(\mu, \sigma^2)$

- μ is known,
- We want to know what is a real guess of σ^2
  - $H_0 : \sigma^2 = \sigma_0^2$ vs $H_1 : \sigma^2 = \sigma_1^2$

Example

Let $x_1, ..., x_n$ be a random variables from $N(\mu, \sigma^2)$

- μ is known,
- We want to know what is a real guess of σ^2
  - $H_0 : \sigma^2 = \sigma_0^2$ vs $H_1 : \sigma^2 = \sigma_1^2$

The likelihood for this set of normally distributed data is

$$L\left(\sigma^2|x\right) \propto \frac{1}{\left(\sigma^2\right)^{-\frac{n}{2}}} \exp\left\{-\frac{\sum_{i=1}^{n}\left(x-\mu\right)^2}{2\sigma^2}\right\}$$

The Ratio - remember σ_0^2 and σ_1^2 are constants

Thus

$$\Lambda(\mathbf{x}) = \frac{L\left(\sigma_0^2 \mid \mathbf{x}\right)}{L\left(\sigma_1^2 \mid \mathbf{x}\right)} = \left(\frac{\sigma_0^2}{\sigma_1^2}\right)^{-n/2} \exp\left\{-\frac{1}{2}(\sigma_0^{-2} - \sigma_1^{-2}) \sum_{i=1}^{n} (x_i - \mu)^2\right\}$$

The Ratio - remember σ_0^2 and σ_1^2 are constants

Thus

$$\Lambda(\mathbf{x}) = \frac{L\left(\sigma_0^2 \mid \mathbf{x}\right)}{L\left(\sigma_1^2 \mid \mathbf{x}\right)} = \left(\frac{\sigma_0^2}{\sigma_1^2}\right)^{-n/2} \exp\left\{-\frac{1}{2}(\sigma_0^{-2} - \sigma_1^{-2})\sum_{i=1}^{n}(x_i - \mu)^2\right\}$$

Or basically

$$\Lambda(\mathbf{x}) \propto \exp\left\{-\frac{1}{2}(\sigma_0^{-2} - \sigma_1^{-2})\sum_{i=1}^{n}(x_i - \mu)^2\right\}$$

Thus

If $\sigma_1^2 > \sigma_0^2 \Rightarrow \sigma_0^{-2} - \sigma_1^{-2} > 0$

- Then, $\Lambda(\mathbf{x})$ is a decreasing function because $-\frac{1}{2}(\sigma_0^{-2} - \sigma_1^{-2}) < 0$
  - So if $\sum_{i=1}^{n}(x_i - \mu)^2$ is large enough we should reject H_0

P-value

Definition

- In null-hypothesis significance testing, the p-value is the probability of obtaining test results at least as extreme as the result actually observed, under the assumption that the null hypothesis is correct

P-value

Definition

- In null-hypothesis significance testing, the p-value is the probability of obtaining test results at least as extreme as the result actually observed, under the assumption that the null hypothesis is correct

Something Notable

- A very small p-value means that such an extreme observed outcome would be very unlikely under the null hypothesis.
- American Statistical Association (ASA) said
  - "p-values do not measure the probability that the studied hypothesis is true, or the probability that the data were produced by random chance alone and p-value, or statistical significance, does not measure the size of an effect or the importance of a result"

Then

The rejection threshold depends on several factors

- $\sum_{i=1}^{n}(x_i - \mu)^2$ can be shown to be a scaled Chi-square distributed random variable
  - Actually the p-value is found by using the α (Significance level) , degrees of freedom and the table for Chisquare
    - ★ $\alpha > p - value$ you reject H_0
    - ★ $\alpha < p - value$ you fail to reject H_0

Thus, you have the following steps

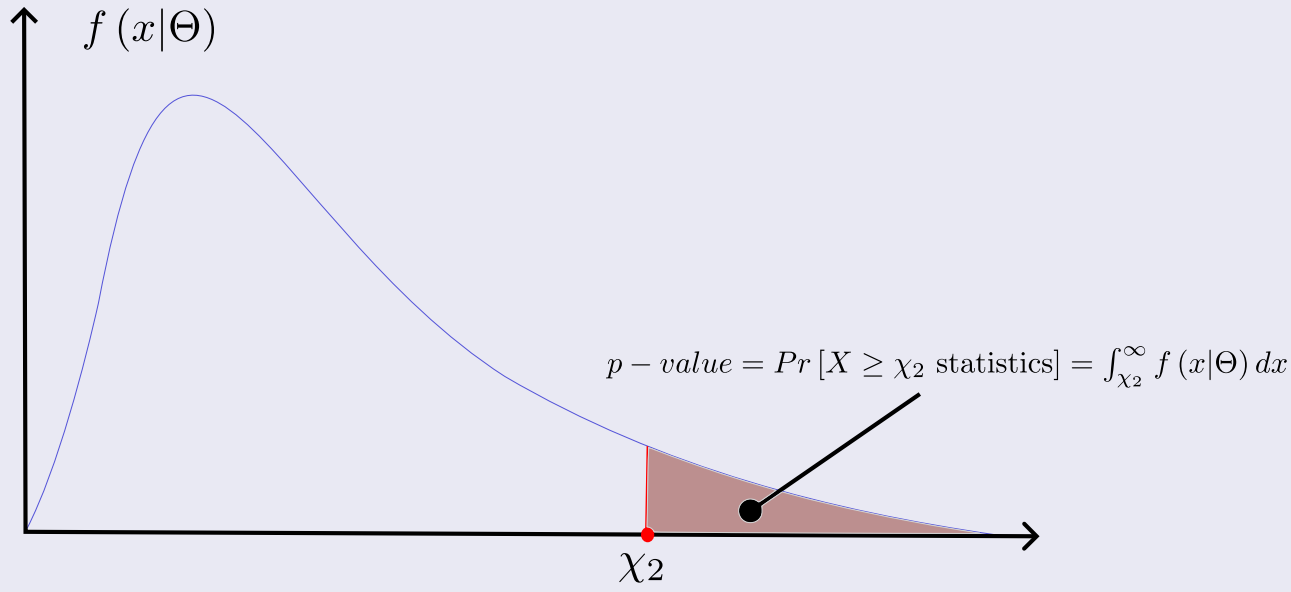
Here are the steps and considerations:

1. Calculate the Test Statistic : In our case $\sum_{i=1}^{n}(x_i-\mu)^2$
2. Determine Degrees of Freedom (df): In our case $n-1$
3. Compute the p-value : The p-value is the probability of observing a test statistic as extreme as, or more extreme than, the one calculated from your sample data

$$p-value=\int_{\sum_{i=1}^{n}(x_i-\mu)^2}^{\infty} f\left(x|\Theta\right)dx=\int_{\sum_{i=1}^{n}(x_i-\mu)^2}^{\infty}\frac{1}{2^{n/2}\Gamma\left(n/2\right)}x^{\left(\frac{n}{2}\right)-1}e^{-x/2}dx$$

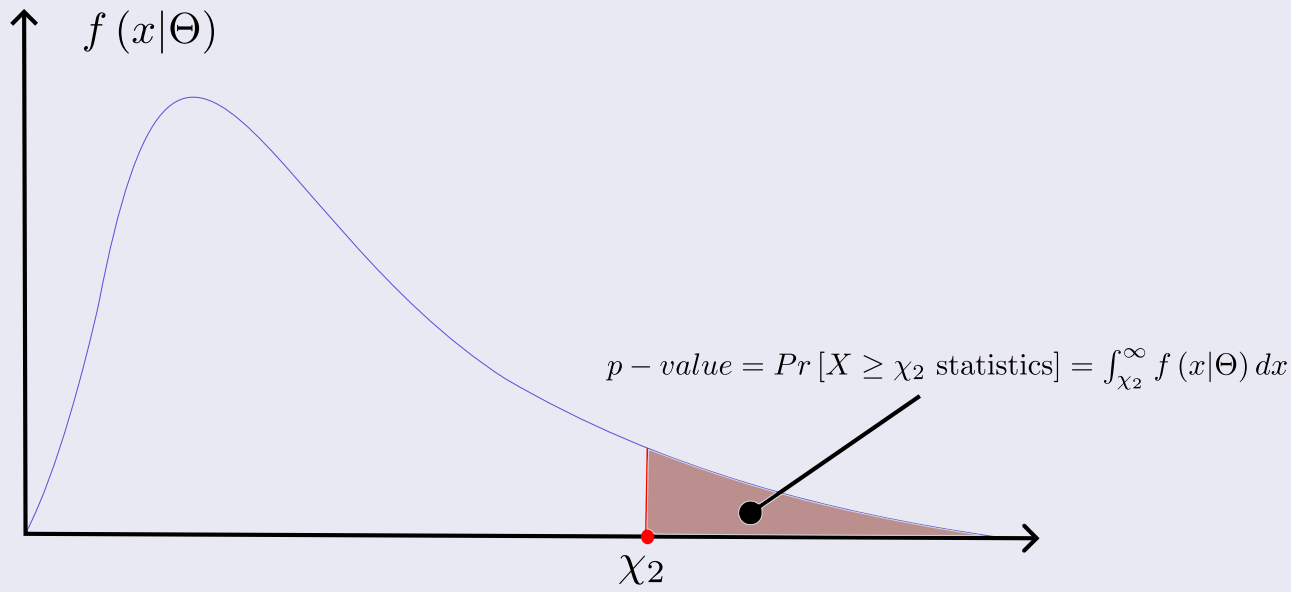
Actually

We have



Actually

We have



Threfore based in α

- $\alpha > p-value$ you reject H_0
- $\alpha < p-value$ you fail to reject H_0

We could use chi2.sf in scipy for this

Actually, we have that chi2.sf is the survival function or

```
# Import necessary libraries
import numpy as np
from scipy.stats import chi2

# Example observed frequencies
# Assume equal probability
observed = [10, 15, 8, 20]

# Calculate expected frequencies
expected = np.array(observed).mean()

# Degrees of freedom
df = len(observed) - 1

# Calculate the Chi-Square statistic
chi2_statistic = sum((np.array(observed) - expected)**2 / expected)

# Calculate the p-value
p_value = chi2.sf(chi2_statistic, df)

print(f"Chi-Square Statistic: {chi2_statistic}")
print(f"P-value: {p_value}")
```

What? Survival Function = sf

Definition

- Let the lifetime X be a continuous random variable describing the time to failure. If X has cumulative function $F(x)$ and probability density function $f(x)$ on the interval $[0, -\infty)$, then the survival function or reliability function is:

$$S(x) = P(X \geq x) = 1 - F(x) = 1 - \int_0^t f(u)\, du$$

Example with $\alpha = 0.01$

Or How to be unable to say if we have a fair coin!!!

- To determine the likelihood of obtaining 15 or more heads, or 5 or fewer tails in a series of coin flips assuming the coin is fair, you can use the binomial distribution.

Example with $\alpha = 0.01$

Or How to be unable to say if we have a fair coin!!!

- To determine the likelihood of obtaining 15 or more heads, or 5 or fewer tails in a series of coin flips assuming the coin is fair, you can use the binomial distribution.

Assume you are flipping the coin $n = 20$ times

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

Example with $\alpha = 0.01$

Or How to be unable to say if we have a fair coin!!!

- To determine the likelihood of obtaining 15 or more heads, or 5 or fewer tails in a series of coin flips assuming the coin is fair, you can use the binomial distribution.

Assume you are flipping the coin $n = 20$ times

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

where

- X is the number of heads,
- n is the total number of trials (coin flips),
- k is the number of successful outcomes (heads),
- $\binom{n}{k} = \frac{n!}{k!(n-k)!}$ is the binomial coefficient.

Now our probability

The total probability we are interested in is

$$p - value = P(X \geq 15) = \sum_{n=15}^{20} p\left(X = n\right) \approx 0.0207$$

Now our probability

The total probability we are interested in is

$$p - value = P(X \geq 15) = \sum_{n=15}^{20} p\left(X = n\right) \approx 0.0207$$

Problem

- $0.01 < 0.0207$ you fail to reject H_0

Outline

1. Bias-Variance Dilemma
   - Introduction
   - Measuring the difference between optimal and learned
   - The Bias-Variance
   - "Extreme" Example
2. Confusion Matrix
   - Introduction
   - Statistical Testing
   - The α and β errors
   - The Initial Confusion Matrix
     - Metrics from the Confusion Matrix
   - The Multi Class Problem
   - Application in Computer Vision
   - Precision-Recall Curve
     - Application: Average Precision
   - Precision and Recall in the BERT Language Model
3. Receiver Operator Curves (ROC)
   - Introduction
   - Example
   - Algorithm for the ROC Curve
   - Area Under the Curve (AUC)
   - Other Measures: F_1-Measure
4. Cross Validation
   - Introduction
   - How to choose K
   - Types of Cross Validation
   - Solving The Imbalanced Class Problem

α error

Definition (Type I Error - False Positive)

α is the probability that the test will lead to the rejection of the hypothesis H_0 when that hypothesis is true.

α error

Definition (Type I Error - False Positive)

α is the probability that the test will lead to the rejection of the hypothesis H_0 when that hypothesis is true.

Example

1. H_0 : "You have a device that produce circuits with no error"

α error

Definition (Type I Error - False Positive)

α is the probability that the test will lead to the rejection of the hypothesis H_0 when that hypothesis is true.

Example

1. H_0 : "You have a device that produce circuits with no error"
2. You have a device that fails $\alpha = 0.05$ meaning that it fails 5 of the time.

α error

Definition (Type I Error - False Positive)

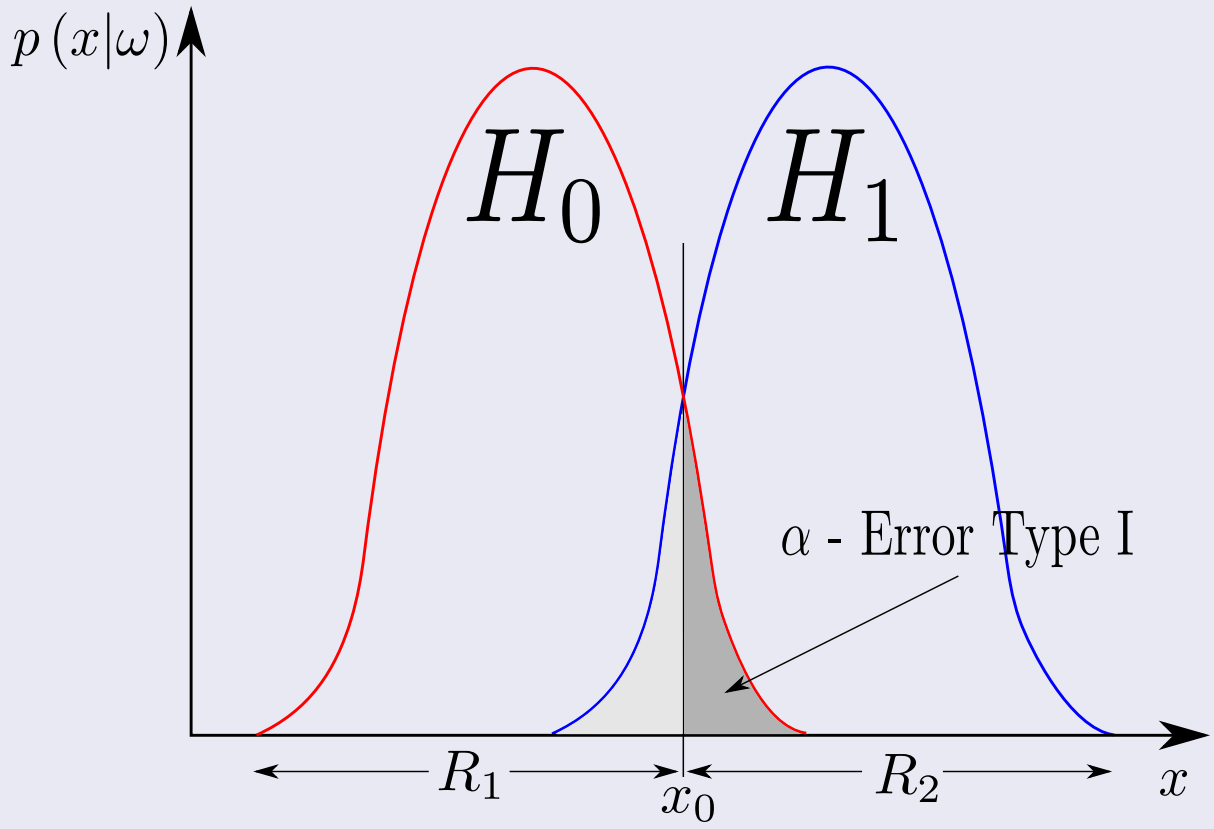
α is the probability that the test will lead to the rejection of the hypothesis H_0 when that hypothesis is true.

Example

1. H_0 : "You have a device that produce circuits with no error"
2. You have a device that fails $\alpha = 0.05$ meaning that it fails 5 of the time.
3. This says that you ha low chance of a wrong circuit.

Basically

We have



β error

Definition (Type II Error - False Negative)

β is the probability that the test will lead to the rejection of the hypothesis H_1 when that hypothesis is true.

β error

Definition (Type II Error - False Negative)

β is the probability that the test will lead to the rejection of the hypothesis H_1 when that hypothesis is true.

Example

1. H_1: “Adding fluoride to toothpaste protects against cavities."

β error

Definition (Type II Error - False Negative)

β is the probability that the test will lead to the rejection of the hypothesis H_1 when that hypothesis is true.

Example

1. H_1: “Adding fluoride to toothpaste protects against cavities."
2. Then $\beta = 0.05$ meaning that you have a chance of 5 of the time.

β error

Definition (Type II Error - False Negative)

β is the probability that the test will lead to the rejection of the hypothesis H_1 when that hypothesis is true.

Example

1. H_1: “Adding fluoride to toothpaste protects against cavities."
2. Then $\beta = 0.05$ meaning that you have a chance of 5 of the time.
3. This says that you have a low chance of having a cavity using fluoride in the water.

Outline

1. Bias-Variance Dilemma
   - Introduction
   - Measuring the difference between optimal and learned
   - The Bias-Variance
   - "Extreme" Example
2. Confusion Matrix
   - Introduction
   - Statistical Testing
   - The α and β errors
   - The Initial Confusion Matrix
     - Metrics from the Confusion Matrix
   - The Multi Class Problem
   - Application in Computer Vision
   - Precision-Recall Curve
     - Application: Average Precision
   - Precision and Recall in the BERT Language Model
3. Receiver Operator Curves (ROC)
   - Introduction
   - Example
   - Algorithm for the ROC Curve
   - Area Under the Curve (AUC)
   - Other Measures: F_1-Measure
4. Cross Validation
   - Introduction
   - How to choose K
   - Types of Cross Validation
   - Solving The Imbalanced Class Problem

This is related to α and β errors

Confusion Matrix

Table of error types		True H_0, False H_1	False H_0, True H_1
Decisions	Reject H_1	Correct Inference **True Positive**	Type II Error - β **False Positive**
	Reject H_0	Type I Error - α **False Negative**	Correct Inference **True Negative**

In the case of two classes, we have

We have finally the Confusion Matrix

		Actual Class	
		Positive	Negative
Predicted Classes	Positive	**True Positive (TP)**	**False Positives (FP)**
	Negative	**False Negatives (FN)**	**True Negatives (TN)**

First, we have the following

We have the following scenario

- Given a sample of 12 individuals,
  - 8 that have been diagnosed with cancer
  - 4 that are cancer-free,

  where individuals with cancer belong to class 1 (positive) and non-cancer individuals belong to class 2 (negative)

First, we have the following

We have the following scenario

- Given a sample of 12 individuals,
  - 8 that have been diagnosed with cancer
  - 4 that are cancer-free,

  where individuals with cancer belong to class 1 (positive) and non-cancer individuals belong to class 2 (negative)

We are completely sure of the following

Identification Number	1	2	3	4	5	6	7	8	9	10	11	12
Real Classification	1	1	1	1	1	1	1	1	2	2	2	2

Now, you have a classification algorithm

We have the following new table

Identification Number	1	2	3	4	5	6	7	8	9	10	11	12
Real Classification	1	1	1	1	1	1	1	1	2	2	2	2
Predicted Classification	2	2	1	1	1	1	1	1	1	2	2	2

Now, you have a classification algorithm

We have the following new table

Identification Number	1	2	3	4	5	6	7	8	9	10	11	12
Real Classification	1	1	1	1	1	1	1	1	2	2	2	2
Predicted Classification	2	2	1	1	1	1	1	1	1	2	2	2

We have the labels

Identification Number	1	2	3	4	5	6	7	8	9	10	11	12
Real Classification	1	1	1	1	1	1	1	1	2	2	2	2
Predicted Classification	2	2	1	1	1	1	1	1	1	2	2	2
Labeling	FN	FN	TP	TP	TP	TP	TP	TP	FP	TN	TN	TN

Then, you count

You generate a table

Table of error types		True H_0, False H_1	False H_0, True H_1
Decisions	Reject H_1	6 **True Positive**	1 **False Positive**
	Reject H_0	2 **False Negative**	3 **True Negative**

Outline

1. Bias-Variance Dilemma
   - Introduction
   - Measuring the difference between optimal and learned
   - The Bias-Variance
   - "Extreme" Example
2. Confusion Matrix
   - Introduction
   - Statistical Testing
   - The α and β errors
   - The Initial Confusion Matrix
     - Metrics from the Confusion Matrix
   - The Multi Class Problem
   - Application in Computer Vision
   - Precision-Recall Curve
     - Application: Average Precision
   - Precision and Recall in the BERT Language Model
3. Receiver Operator Curves (ROC)
   - Introduction
   - Example
   - Algorithm for the ROC Curve
   - Area Under the Curve (AUC)
   - Other Measures: F_1-Measure
4. Cross Validation
   - Introduction
   - How to choose K
   - Types of Cross Validation
   - Solving The Imbalanced Class Problem

Accuracy

Definition

- The proportion of getting correct classification of the Positive and Negative classes.

Accuracy

Definition

- The proportion of getting correct classification of the Positive and Negative classes.

Thus

$$\text{Accuracy} = \frac{TP + TN}{TP + FN + FP + TN} = \frac{TP + TN}{P + N}$$

		Actual Class	
		Positive	Negative
Predicted Classes	Positive	**True Positive (TP)**	**False Positives (FP)**
	Negative	**False Negatives (FN)**	**True Negatives (TN)**

Accuracy

Definition

- The proportion of getting correct classification of the Positive and Negative classes.

Thus

$$\text{Accuracy} = \frac{TP + TN}{TP + FN + FP + TN} = \frac{TP + TN}{P + N}$$

		Actual Class	
		Positive	Negative
Predicted Classes	Positive	**True Positive (TP)**	**False Positives (FP)**
	Negative	**False Negatives (FN)**	**True Negatives (TN)**

Problem - accuracy assumes equal cost for both kinds of errors

Is 99% accuracy good, bad or terrible? It depends on the problem.

Another Problem

You have only a way to measure the total correct answers

- Yes, we have at the numerator of $TP + TN$

Another Problem

You have only a way to measure the total correct answers

- Yes, we have at the numerator of $TP + TN$

We need to measure for the Positive Class/Class 1

- Yes, we need to measure the moments when the correct answers can be correct

Another Problem

You have only a way to measure the total correct answers

- Yes, we have at the numerator of $TP + TN$

We need to measure for the Positive Class/Class 1

- Yes, we need to measure the moments when the correct answers can be correct

The True Positive or

- The Recall Rate...

True Positive Rate

Also called

- Sensitivity or **Recall Rate**

True Positive Rate

Also called

- Sensitivity or **Recall Rate**

Defined as

- True Positive Rate is the proportion of getting a correct classification of the Positive Class vs the True Positive and False Negatives.

$$\text{Recall} = \frac{TP}{TP + FN} = \frac{TP}{P}$$

		Actual Class	
		Positive	Negative
Predicted Classes	Positive	**True Positive (TP)**	False Positives (FP)
	Negative	**False Negatives (FN)**	True Negatives (TN)

We also need to measure the Class 2

For this, we have

- Specificity

True Negative Rate

Also known as

- Specificity

True Negative Rate

Also known as

- Specificity

Defined as

- It is the proportion of True Negative vs the elements classified as True negatives.

$$\text{Specifity} = \frac{TN}{FP + TN} = \frac{TN}{N}$$

		Actual Class	
		Positive	Negative
Predicted Classes	Positive	True Positive (TP)	**False Positives (FP)**
	Negative	False Negatives (FN)	**True Negatives (TN)**

We also need to measure the rate of TP

We have for this

- The Precision

Precision

Also known as

- **Positive Predictive Value**

Precision

Also known as

- **Positive Predictive Value**

Defined as

- The proportion of the elements classified as true positive vs the total of all the real true positives.

$$\text{Precision} = \frac{TP}{FP + TP}$$

		Actual Class	
		Positive	Negative
Predicted Classes	Positive	**True Positive (TP)**	**False Positives (FP)**
	Negative	False Negatives (FN)	True Negatives (TN)

Significance Level

Also known as

False Positive Rate.

Significance Level

Also known as

False Positive Rate.

Defined as

False Positive Rate is the probability of getting an incorrect classification of the Positive Class vs the True Negative and the False Positive.

$$\text{False positive rate} = \frac{FP}{TN + FP}$$

		Actual Class	
		Positive	Negative
Predicted Classes	Positive	True Positive (TP)	**False Positives (FP)**
	Negative	False Negatives (FN)	**True Negatives (TN)**

Outline

1. Bias-Variance Dilemma
   - Introduction
   - Measuring the difference between optimal and learned
   - The Bias-Variance
   - "Extreme" Example
2. Confusion Matrix
   - Introduction
   - Statistical Testing
   - The α and β errors
   - The Initial Confusion Matrix
     - Metrics from the Confusion Matrix
   - The Multi Class Problem
   - Application in Computer Vision
   - Precision-Recall Curve
     - Application: Average Precision
   - Precision and Recall in the BERT Language Model
3. Receiver Operator Curves (ROC)
   - Introduction
   - Example
   - Algorithm for the ROC Curve
   - Area Under the Curve (AUC)
   - Other Measures: F_1-Measure
4. Cross Validation
   - Introduction
   - How to choose K
   - Types of Cross Validation
   - Solving The Imbalanced Class Problem

We need to test the multiclass classifier

Once we have a classifier, say ball sizes

- We can build a confusion matrix for balls

	$Small$	$Medium$	$Large$	$X\text{-}Large$
$Small$	**15**	2	2	0
$Medium$	3	**20**	2	0
$Large$	2	3	**10**	3
$X\text{-}Large$	1	1	0	**5**

Therefore

We have recall and precision

	$Small$	$Medium$	$Large$	$X\text{-}Large$	$Precision$
$Small$	$\mathbf{15}$	2	2	0	$\frac{15}{15+2+2+0}$
$Medium$	3	$\mathbf{20}$	2	0	$\frac{20}{3+20+2+0}$
$Large$	2	3	$\mathbf{10}$	3	$\frac{10}{2+3+10+3}$
$X\text{-}Large$	1	1	0	$\mathbf{5}$	$\frac{5}{1+1+0+5}$
$Recall$	$\frac{15}{15+3+2+1}$	$\frac{20}{2+20+3+1}$	$\frac{10}{2+2+10+0}$	$\frac{5}{0+0+3+5}$	

We have a better options

Micro Averages

- It gives equal weight to every instance and shows average performance across all predictions.

We have a better options

Micro Averages

- It gives equal weight to every instance and shows average performance across all predictions.

Macro Averages

- It shows average performance across classes, treating each class as equally important.

We can do the following

Small Ball Confusion Matrix

	Actual Small	Actual Not Small
Pred Small	15	4
Pred Not Small	6	44

We can do the following

Small Ball Confusion Matrix

	Actual Small	Actual Not Small
Pred Small	15	4
Pred Not Small	6	44

Medium Ball Confusion Matrix

	Actual Medium	Actual Not Medium
Pred Medium	20	4
Pred Not Medium	6	39

Furthermore

Large Ball Confusion Matrix

	Actual Large	Actual Not Large
Pred Large	10	8
Pred Not Large	4	47

Furthermore

Large Ball Confusion Matrix

	Actual Large	Actual Not Large
Pred Large	10	8
Pred Not Large	4	47

X-Large Ball Confusion Matrix

	Actual X-Large	Actual Not X-Large
Pred X-Large	5	3
Pred Not X-Large	4	59

Confusion Matrix for the multi-class

C_{ij} representing the entry of the confusion matrix at i^{th} row and j^{th} column

- Then the Recall for the i^{th} class, it represent the fraction of the data-points classified correctly

$$Recall_i = \frac{C_{ii}}{\sum_{j=1}^{M} C_{ji}}$$

Confusion Matrix for the multi-class

C_{ij} representing the entry of the confusion matrix at i^{th} row and j^{th} column

- Then the Recall for the i^{th} class, it represent the fraction of the data-points classified correctly

$$Recall_i = \frac{C_{ii}}{\sum_{j=1}^{M} C_{ji}}$$

Precision of i^{th} class

$$Precision_i = \frac{C_{ii}}{\sum_{j=1}^{M} C_{ij}}$$

Confusion Matrix for the multi-class

C_{ij} representing the entry of the confusion matrix at i^{th} row and j^{th} column

- Then the Recall for the i^{th} class, it represent the fraction of the data-points classified correctly

$$Recall_i = \frac{C_{ii}}{\sum_{j=1}^{M} C_{ji}}$$

Precision of i^{th} class

$$Precision_i = \frac{C_{ii}}{\sum_{j=1}^{M} C_{ij}}$$

Accuracy

$$Accuracy = \frac{\sum_{i=1}^{M} C_{ii}}{\sum_{j=1}^{M} C_{ij}}$$

Furthermore

Once you have calculated each precision

- We can calculate the Macro-Average Precision for example

$$Precision = \frac{Prec_1 + Prec_2 + Prec_3 + Prec_4}{4}$$

We have that we can calculate the micro average

By collecting all the observations

	True	False
$True$	50	19
$False$	20	189

We have that we can calculate the micro average

By collecting all the observations

	True	False
$True$	50	19
$False$	20	189

Micro Average Local

$$\frac{TP}{TP + FN} = \frac{50}{50 + 20}$$

Outline

1. Bias-Variance Dilemma
   - Introduction
   - Measuring the difference between optimal and learned
   - The Bias-Variance
   - "Extreme" Example
2. Confusion Matrix
   - Introduction
   - Statistical Testing
   - The α and β errors
   - The Initial Confusion Matrix
     - Metrics from the Confusion Matrix
   - The Multi Class Problem
   - Application in Computer Vision
   - Precision-Recall Curve
     - Application: Average Precision
   - Precision and Recall in the BERT Language Model
3. Receiver Operator Curves (ROC)
   - Introduction
   - Example
   - Algorithm for the ROC Curve
   - Area Under the Curve (AUC)
   - Other Measures: F_1-Measure
4. Cross Validation
   - Introduction
   - How to choose K
   - Types of Cross Validation
   - Solving The Imbalanced Class Problem

Introduction

In computer vision, we have different scenarios

- For example we have a two class scenario in an image cancer and non cancer

Introduction

In computer vision, we have different scenarios

- For example we have a two class scenario in an image cancer and non cancer

We want to know the classification precision of an algorithm doing semantic segmentation

- What do we do?

We recall Precision

$$\text{Precision Predicted Value} = \frac{TP}{FP + TP}$$

- In this case we have a mask that defines the semantic segmentation so we can calculate how many pixels are correct
  - And which are not correct.

Intersection over Union (IoU) as Precision

What do we need

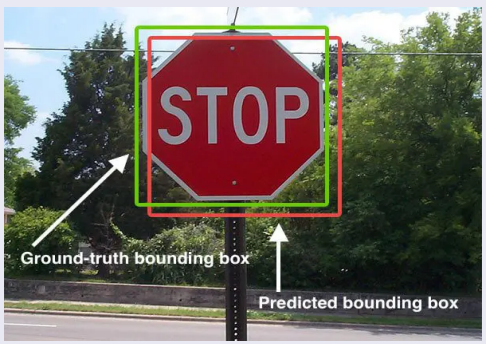
- More formally, in order to apply Intersection over Union to evaluate an (arbitrary) object detector we need:
  - The ground-truth bounding boxes (i.e., the hand labeled bounding boxes from the testing set that specify where in the image our object is).
  - The predicted bounding boxes from our model.

Intersection over Union (IoU) as Precision

What do we need

- More formally, in order to apply Intersection over Union to evaluate an (arbitrary) object detector we need:
  - The ground-truth bounding boxes (i.e., the hand labeled bounding boxes from the testing set that specify where in the image our object is).
  - The predicted bounding boxes from our model.

We have then



$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$

Precision based in this measure

We have

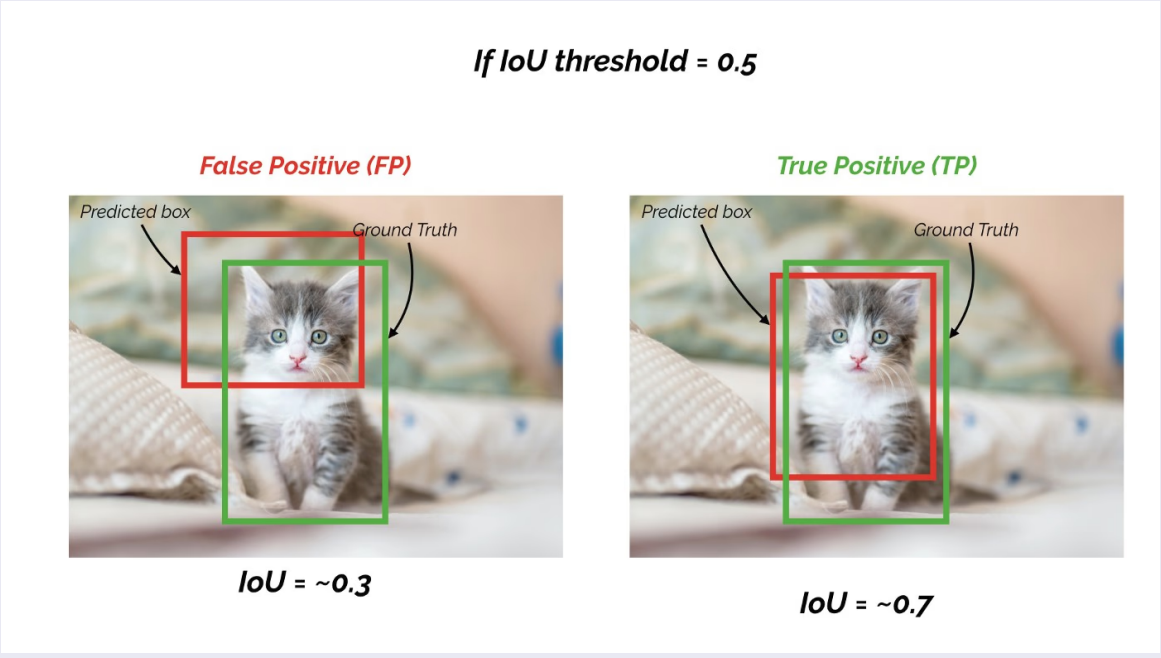
- For instance, the precision is calculated using the IoU threshold in object detection tasks.

Precision based in this measure

We have

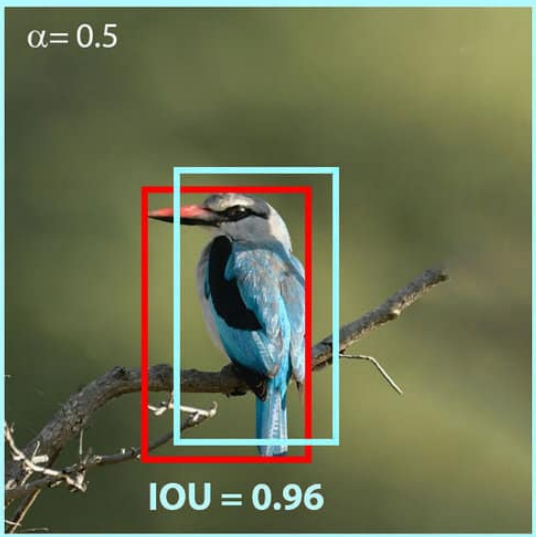
- For instance, the precision is calculated using the IoU threshold in object detection tasks.

We have

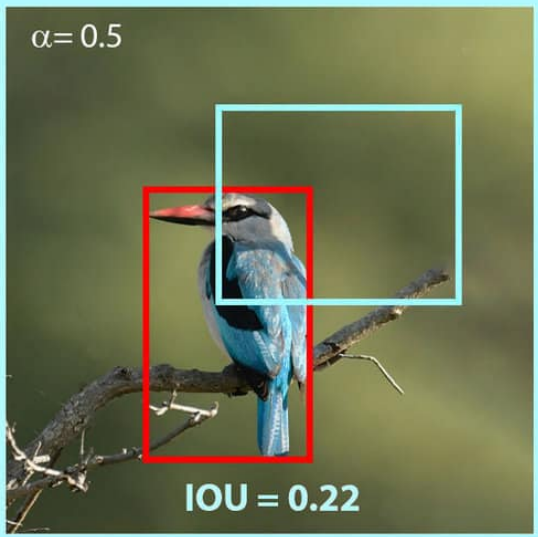


Remarking this...

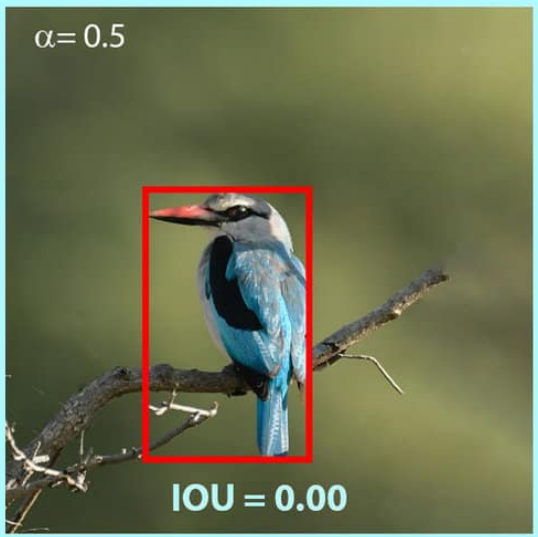
Assuming $\alpha = 0.5$



True Positive



False Positive



False Negative

Outline

1. Bias-Variance Dilemma
   - Introduction
   - Measuring the difference between optimal and learned
   - The Bias-Variance
   - "Extreme" Example
2. Confusion Matrix
   - Introduction
   - Statistical Testing
   - The α and β errors
   - The Initial Confusion Matrix
     - Metrics from the Confusion Matrix
   - The Multi Class Problem
   - Application in Computer Vision
   - Precision-Recall Curve
     - Application: Average Precision
   - Precision and Recall in the BERT Language Model
3. Receiver Operator Curves (ROC)
   - Introduction
   - Example
   - Algorithm for the ROC Curve
   - Area Under the Curve (AUC)
   - Other Measures: F_1-Measure
4. Cross Validation
   - Introduction
   - How to choose K
   - Types of Cross Validation
   - Solving The Imbalanced Class Problem

Precision Recall Curve

Remember

$$Precision = \frac{TP}{FP + TP}, Recall = \frac{TP}{TP + FN}$$

Now, we have

Definition - Precision Recall (PR)

- A PR curve is simply a graph with Precision values on the y-axis and Recall values on the x-axis.

Now, we have

Definition - Precision Recall (PR)

- A PR curve is simply a graph with Precision values on the y-axis and Recall values on the x-axis.

Starting Point

- When the classifier sets a very high threshold for predicting positive instances, it will predict almost everything as negative.
- This results in perfect precision (since no true positives are predicted incorrectly), but zero recall because all actual positives are missed.
- Thus, the curve starts at (0,1) where the x-axis represents recall and the y-axis represents precision.

Then

We have for the final point

- Conversely, when the classifier sets a very low threshold (essentially predicting every instance as positive), it will capture all actual positives but also many negatives.
- This results in perfect recall since no true positives are missed, but lower precision because there will be false positives.
- The end point is calculated as$(P/(P+N), 1.0)$.

Code

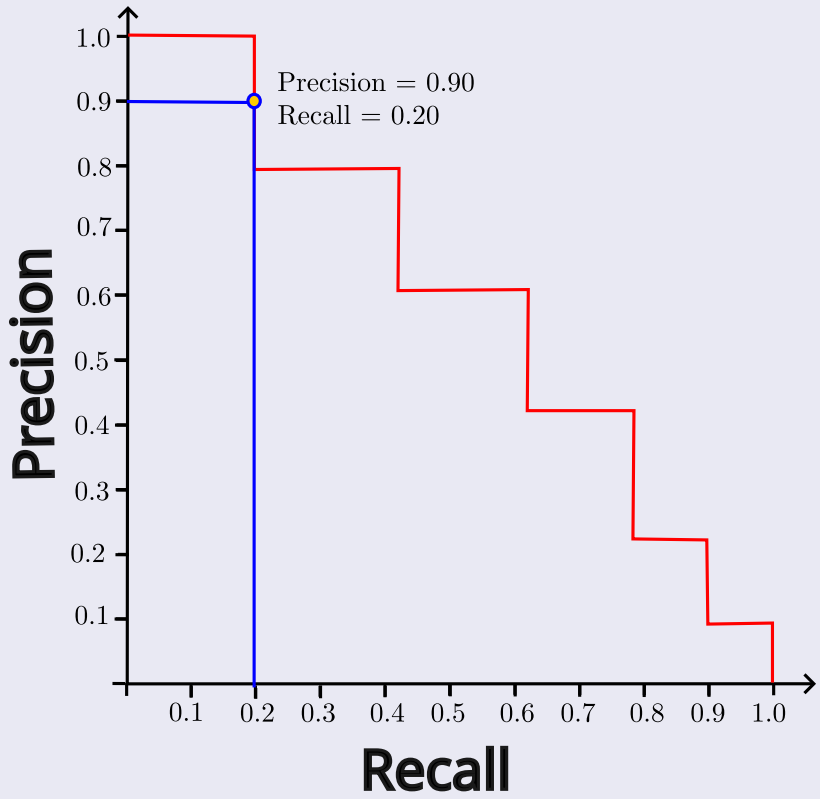
We have the following code

- Input scores by a classifier y_{scores} and the ground truth y_{true}

1. **Get index** s_{index} **from sorting by arg** $-y_{scores}$
2. **Use the indixes to sort** $y_{true-sorted} = y_{true}[s_{index}]$
3. **Get the total number positive elements** num_{pos}
4. $precisions = recalls = []$, $tp_{count} = fp_{count} = 0$
5. **for** idx **in range(len(**y_{scores} **)):**
6. **if** $y_{true-sorted}[idx] == 1$**:** $tp_{count} += 1$ **else:** $fp_{count} += 1$
7. $precision$ **=** tp_{count}**/(**$tp_{count} + fp_{count}$**) if (**$tp_{count} + fp_{count}$**) >0 else 0**
8. $recall$**=** tp_{count}**/** num_{pos}
9. $precisions = precisions \cup precision$, $recalls = recalls \cup recall$

Example

A Good Precision and Bad Recall



Outline

1. Bias-Variance Dilemma
   - Introduction
   - Measuring the difference between optimal and learned
   - The Bias-Variance
   - "Extreme" Example
2. Confusion Matrix
   - Introduction
   - Statistical Testing
   - The α and β errors
   - The Initial Confusion Matrix
     - Metrics from the Confusion Matrix
   - The Multi Class Problem
   - Application in Computer Vision
   - Precision-Recall Curve
     - Application: Average Precision
   - Precision and Recall in the BERT Language Model
3. Receiver Operator Curves (ROC)
   - Introduction
   - Example
   - Algorithm for the ROC Curve
   - Area Under the Curve (AUC)
   - Other Measures: F_1-Measure
4. Cross Validation
   - Introduction
   - How to choose K
   - Types of Cross Validation
   - Solving The Imbalanced Class Problem

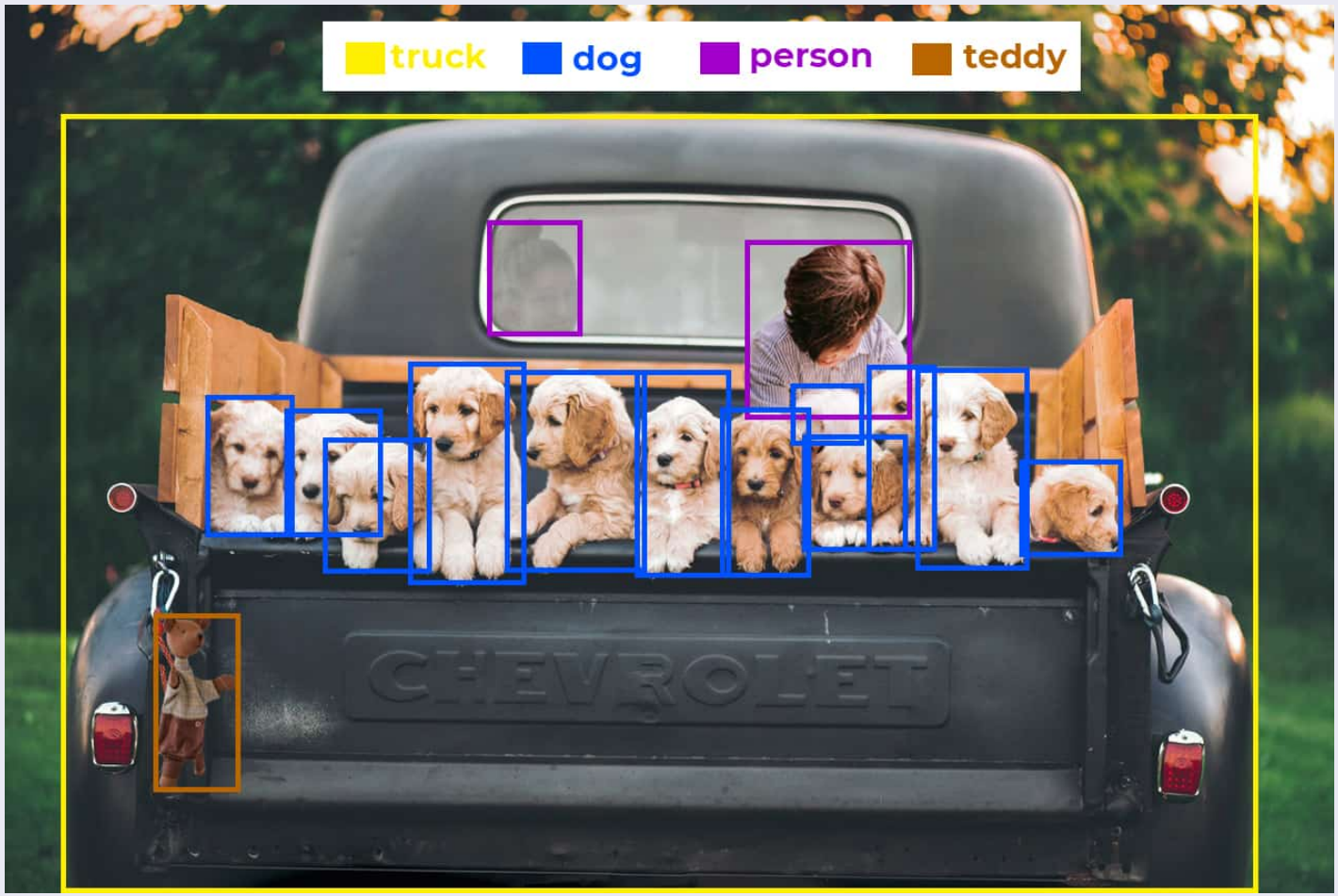
Now we have Average Precision (AP)

Average Precision

- Average Precision (AP) is not the average of Precision (P). The term AP has evolved with time.
- For simplicity, we can say that it is the area under the precision-recall curve.

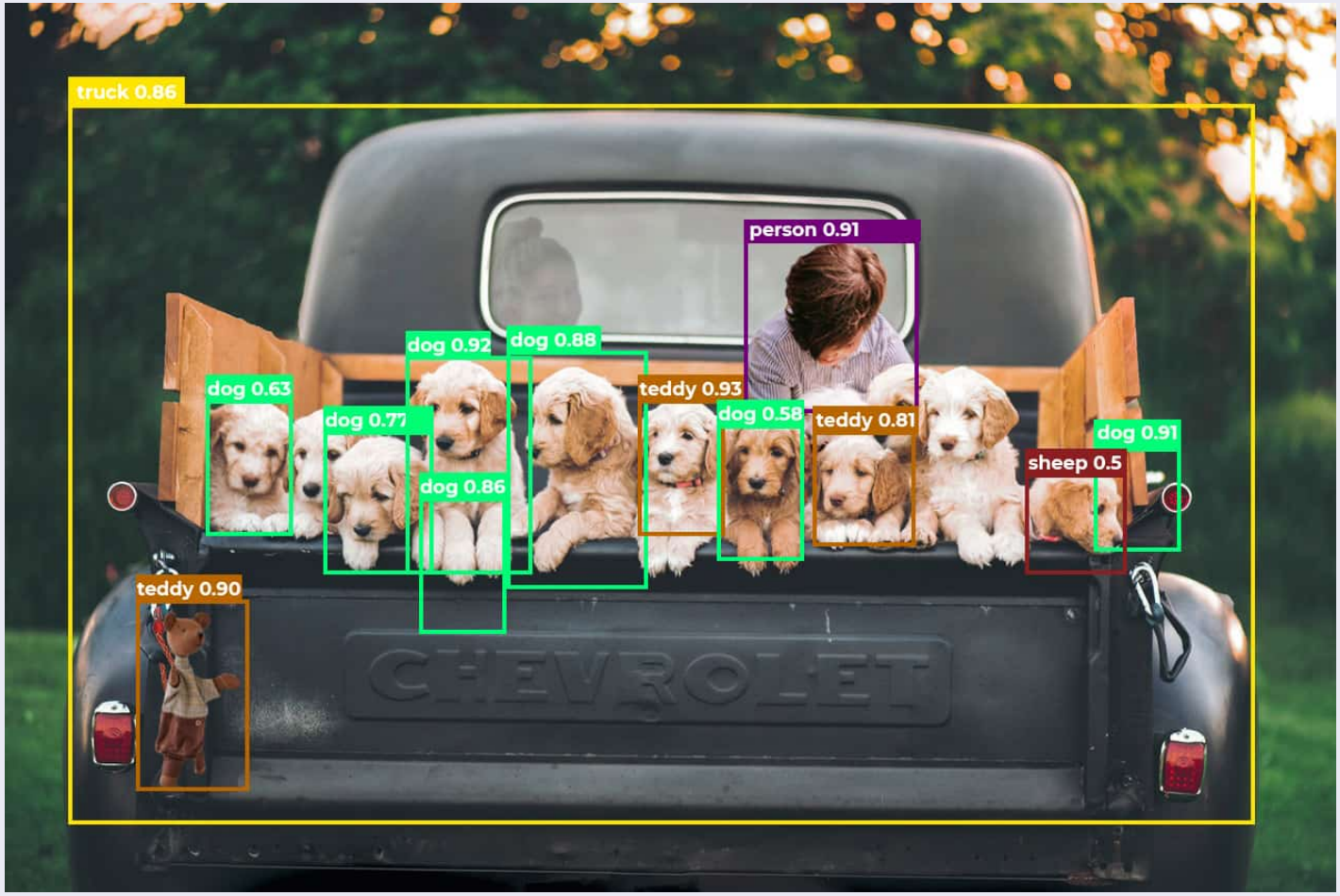
Example

Ground Truth



Then, we have the following

We have the following confidences



Then Labeling

Using IoU

Detections							
Conf.	0.63	0.77	0.92	0.86	0.88	0.58	0.91
Matches GT by IoU?	TP	TP	TP	FP	TP	TP	FP

Remember the Precision and Recall Code

Preds.	Conf.	Matches	Cumulative TP	Cumulative FP	Precision	Recall
	0.92	TP	1	0	1/(1+0) = 1	1/16 = 0.08
	0.91	FP	1	1	1/(1+1) = 0.5	1/16 = 0.08
	0.88	TP	2	1	2/(2+3) = 0.66	2/16 = 0.16
	0.86	FP	2	2	0.5	0.16
	0.77	TP	3	2	0.6	0.25
	0.63	TP	4	2	0.66	0.33
	0.58	TP	5	2	0.71	0.41

Then, we have

The final formula for the Average Precision to be

$$AP = \sum_{i=0}^{n-1} [recall\,[i] - recall\,[i+1]] * precision\,[i]$$

Outline

1. Bias-Variance Dilemma
   - Introduction
   - Measuring the difference between optimal and learned
   - The Bias-Variance
   - "Extreme" Example
2. Confusion Matrix
   - Introduction
   - Statistical Testing
   - The α and β errors
   - The Initial Confusion Matrix
     - Metrics from the Confusion Matrix
   - The Multi Class Problem
   - Application in Computer Vision
   - Precision-Recall Curve
     - Application: Average Precision
   - Precision and Recall in the BERT Language Model
3. Receiver Operator Curves (ROC)
   - Introduction
   - Example
   - Algorithm for the ROC Curve
   - Area Under the Curve (AUC)
   - Other Measures: F_1-Measure
4. Cross Validation
   - Introduction
   - How to choose K
   - Types of Cross Validation
   - Solving The Imbalanced Class Problem

Language Model

Testing a Model f for example BERT

- It involves evaluating its performance on a held-out dataset by feeding it various text samples and measuring
  - accuracy in completing tasks like sentiment analysis, text classification, question answering, or named entity recognition

Language Model

Testing a Model f for example BERT

- It involves evaluating its performance on a held-out dataset by feeding it various text samples and measuring
  - accuracy in completing tasks like sentiment analysis, text classification, question answering, or named entity recognition

Basically for example

- question answering, a series of questions are asked to the Language Model where answers (ground truth) already exist

Language Model

Testing a Model f for example BERT

- It involves evaluating its performance on a held-out dataset by feeding it various text samples and measuring
  - accuracy in completing tasks like sentiment analysis, text classification, question answering, or named entity recognition

Basically for example

- question answering, a series of questions are asked to the Language Model where answers (ground truth) already exist

The results

- They are compared

The beginning of this

Early Tests

- The commonly used techniques for text evaluation are based on n-Gram matching.
- The main objective here is to compare the n-grams in reference and candidate sentences and thus analyze the ordering of words in the sentences.

The beginning of this

Early Tests

- The commonly used techniques for text evaluation are based on n-Gram matching.
- The main objective here is to compare the n-grams in reference and candidate sentences and thus analyze the ordering of words in the sentences.

We finish with the following equations

$$Exact - P_n = \frac{\sum_{w \in S^n_{\widehat{x}}} I\left[w \in S^n_x\right]}{\left|S^n_{\widehat{x}}\right|}$$

$$Exact - R_n = \frac{\sum_{w \in S^n_x} I\left[w \in S^n_{\widehat{x}}\right]}{|S^n_x|}$$

- I is an indicator function,
- X is the ground truth text and $\widehat{X}$ is the generated text by the LLM.
- $S^n_x, S^n_{\widehat{x}}$ are lists of token n-grams in the ground truth and candidate sentences respectively.

Here

The most popular n-Gram Matching metric is BLEU (Bilingual Evaluation Understudy)

- The output for this metric is between 0.0 and 1.0 where a score of 0.0 denotes a perfect mismatch and a score of 1.0 denotes a perfect match between candidate sentence and reference sentence.

Problems!!!

What happened when you have something like this

- Ground Truth: people like foreign cars
  - Candidate 1: people like visiting places abroad
  - Candidate 2: consumers prefer imported cars

Problems!!!

What happened when you have something like this

- Ground Truth: people like foreign cars
  - Candidate 1: people like visiting places abroad
  - Candidate 2: consumers prefer imported cars

Something Notable

- BLEU gives a higher score to Candidate 1 as compared to Candidate 2.

BERT score tries to overcome this problem

Here, we have X tokens at ground truth and $\widehat{X}$ candidate sentence generated by BERT

- Pairwise cosine similarity is calculated between each token x_i in ground truth sentence and $\widehat{x}_j$ in candidate sentence.
- Prenormalized vectors are used, therefore the pairwise similarity is given by $x_i^T \widehat{x}_j$

BERT score tries to overcome this problem

Here, we have X tokens at ground truth and $\widehat{X}$ candidate sentence generated by BERT

- Pairwise cosine similarity is calculated between each token x_i in ground truth sentence and $\widehat{x}_j$ in candidate sentence.
- Prenormalized vectors are used, therefore the pairwise similarity is given by $x_i^T \widehat{x}_j$

Recall BERT

$$R_{BERT} = \frac{1}{|X|} \sum_{x_i \in X} \max_{\widehat{x}_j \in \widehat{X}} x_i^T \widehat{x}_j$$

Next

Precision BERT

$$P_{BERT} = \frac{1}{\left|\widehat{X}\right|} \sum_{\widehat{x}_j \in \widehat{X}} \max_{x_i \in X} x_i^T \widehat{x}_j$$

Why is this?

We need to explain some phenomena happening at embedding

- Look at the Board

Outline

1. Bias-Variance Dilemma
   - Introduction
   - Measuring the difference between optimal and learned
   - The Bias-Variance
   - "Extreme" Example
2. Confusion Matrix
   - Introduction
   - Statistical Testing
   - The α and β errors
   - The Initial Confusion Matrix
     - Metrics from the Confusion Matrix
   - The Multi Class Problem
   - Application in Computer Vision
   - Precision-Recall Curve
     - Application: Average Precision
   - Precision and Recall in the BERT Language Model
3. Receiver Operator Curves (ROC)
   - Introduction
   - Example
   - Algorithm for the ROC Curve
   - Area Under the Curve (AUC)
   - Other Measures: F_1-Measure
4. Cross Validation
   - Introduction
   - How to choose K
   - Types of Cross Validation
   - Solving The Imbalanced Class Problem

We can do better than these simple measures of accuracy

Given these initial measures of validity

it is possible to obtain a more precise model evaluation, the ROC curves.

We can do better than these simple measures of accuracy

Given these initial measures of validity

it is possible to obtain a more precise model evaluation, the ROC curves.

The ROC Curves plot

It is a model wide evaluation measure that is based on two basic evaluation measures:

1. **Specificity** is a performance measure of the whole negative part of a dataset.
2. **Sensitivity** is a performance measure of the whole positive part.

What the ROC Curves uses

We have a plot where

The ROC plot uses specificity on the x-axis and sensitivity on the y-axis.

What the ROC Curves uses

We have a plot where

The ROC plot uses specificity on the x-axis and sensitivity on the y-axis.

Basically

False Positive Rate (FPR) is identical with specificity, and True Positive Rate (TPR) is identical with sensitivity.

$$\text{Specificity} = \text{False positive rate} = \frac{FP}{TN + FP}$$

$$\text{True Positive Rate} = \frac{TP}{TP + FN}$$

What the ROC Curves uses

We have a plot where

The ROC plot uses specificity on the x-axis and sensitivity on the y-axis.

Basically

False Positive Rate (FPR) is identical with specificity, and True Positive Rate (TPR) is identical with sensitivity.

$$\text{Specificity} = \text{False positive rate} = \frac{FP}{TN + FP}$$

$$\text{True Positive Rate} = \frac{TP}{TP + FN}$$

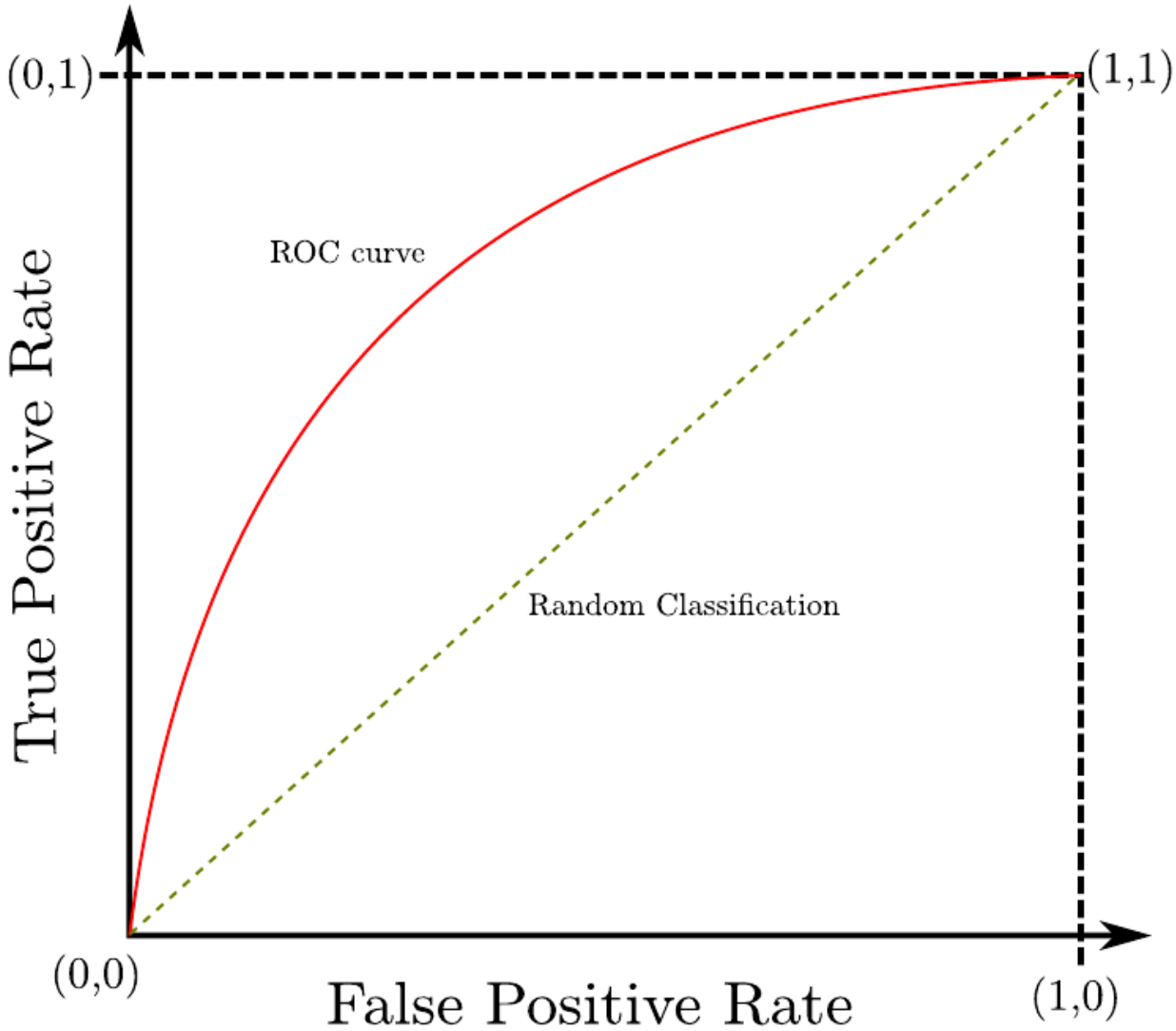
Then

1. A ROC curve is created by connecting all ROC points of a classier in the ROC space.
2. Two adjacent ROC points can be connected by a straight line.
3. The curve starts at (0.0, 0.0) and ends at (1.0, 1.0).

Outline

1. Bias-Variance Dilemma
   - Introduction
   - Measuring the difference between optimal and learned
   - The Bias-Variance
   - "Extreme" Example
2. Confusion Matrix
   - Introduction
   - Statistical Testing
   - The α and β errors
   - The Initial Confusion Matrix
     - Metrics from the Confusion Matrix
   - The Multi Class Problem
   - Application in Computer Vision
   - Precision-Recall Curve
     - Application: Average Precision
   - Precision and Recall in the BERT Language Model
3. Receiver Operator Curves (ROC)
   - Introduction
   - Example
   - Algorithm for the ROC Curve
   - Area Under the Curve (AUC)
   - Other Measures: F_1-Measure
4. Cross Validation
   - Introduction
   - How to choose K
   - Types of Cross Validation
   - Solving The Imbalanced Class Problem

For Example


(0,1)
(1,1)
ROC curve
Random Classification
True Positive Rate
(0,0)
False Positive Rate
(1,0)

Outline

1. Bias-Variance Dilemma
   - Introduction
   - Measuring the difference between optimal and learned
   - The Bias-Variance
   - "Extreme" Example
2. Confusion Matrix
   - Introduction
   - Statistical Testing
   - The α and β errors
   - The Initial Confusion Matrix
     - Metrics from the Confusion Matrix
   - The Multi Class Problem
   - Application in Computer Vision
   - Precision-Recall Curve
     - Application: Average Precision
   - Precision and Recall in the BERT Language Model
3. Receiver Operator Curves (ROC)
   - Introduction
   - Example
   - Algorithm for the ROC Curve
   - Area Under the Curve (AUC)
   - Other Measures: F_1-Measure
4. Cross Validation
   - Introduction
   - How to choose K
   - Types of Cross Validation
   - Solving The Imbalanced Class Problem

We have

Algorithm ROC point generation

Input: L, the set of test examples; $f(i)$, the probabilistic classifier estimate that example i is positive; P and N, the number of positive and negative examples.

We have

Algorithm ROC point generation

Input: L, the set of test examples; $f(i)$, the probabilistic classifier estimate that example i is positive; P and N, the number of positive and negative examples.

Ouput: R, a list of ROC points increasing by false positive rate.

We have

Algorithm ROC point generation

Input: L, the set of test examples; $f(i)$, the probabilistic classifier estimate that example i is positive; P and N, the number of positive and negative examples.

Ouput: R, a list of ROC points increasing by false positive rate.

1. $L_{sorted} \leftarrow L$ **sorted decreasing by** f **scores**

We have

Algorithm ROC point generation

Input: L, the set of test examples; $f(i)$, the probabilistic classifier estimate that example i is positive; P and N, the number of positive and negative examples.

Ouput: R, a list of ROC points increasing by false positive rate.

1. $L_{sorted} \leftarrow L$ **sorted decreasing by** f **scores**
2. $FP \leftarrow TP \leftarrow 0; R \leftarrow \langle\rangle; fprev \leftarrow -\infty; i \leftarrow 1$

We have

Algorithm ROC point generation

Input: L, the set of test examples; $f(i)$, the probabilistic classifier estimate that example i is positive; P and N, the number of positive and negative examples.

Ouput: R, a list of ROC points increasing by false positive rate.

1. $L_{sorted} \leftarrow L$ **sorted decreasing by** f **scores**
2. $FP \leftarrow TP \leftarrow 0; R \leftarrow \langle\rangle; fprev \leftarrow -\infty; i \leftarrow 1$
3. **while** $i \leq |L_{sorted}|$

We have

Algorithm ROC point generation

Input: L, the set of test examples; $f(i)$, the probabilistic classifier estimate that example i is positive; P and N, the number of positive and negative examples.

Ouput: R, a list of ROC points increasing by false positive rate.

1. $L_{sorted} \leftarrow L$ **sorted decreasing by** f **scores**
2. $FP \leftarrow TP \leftarrow 0; R \leftarrow \langle\rangle; fprev \leftarrow -\infty; i \leftarrow 1$
3. **while** $i \leq |L_{sorted}|$
4. **if** $f(i) \neq f_{prev}$ **then**

We have

Algorithm ROC point generation

Input: L, the set of test examples; $f(i)$, the probabilistic classifier estimate that example i is positive; P and N, the number of positive and negative examples.

Ouput: R, a list of ROC points increasing by false positive rate.

1. $L_{sorted} \leftarrow L$ **sorted decreasing by** f **scores**
2. $FP \leftarrow TP \leftarrow 0; R \leftarrow \langle\rangle; fprev \leftarrow -\infty; i \leftarrow 1$
3. **while** $i \leq |L_{sorted}|$
4. **if** $f(i) \neq f_{prev}$ **then**
5. $R.append\left(\frac{FP}{N}, \frac{TP}{P}\right)$

We have

Algorithm ROC point generation

Input: L, the set of test examples; $f(i)$, the probabilistic classifier estimate that example i is positive; P and N, the number of positive and negative examples.

Ouput: R, a list of ROC points increasing by false positive rate.

1. $L_{sorted} \leftarrow L$ **sorted decreasing by** f **scores**
2. $FP \leftarrow TP \leftarrow 0; R \leftarrow \langle\rangle; fprev \leftarrow -\infty; i \leftarrow 1$
3. **while** $i \leq |L_{sorted}|$
4. **if** $f(i) \neq f_{prev}$ **then**
5. $R.append\left(\frac{FP}{N}, \frac{TP}{P}\right)$
6. $fprev \leftarrow f(i)$

We have

Algorithm ROC point generation

Input: L, the set of test examples; $f(i)$, the probabilistic classifier estimate that example i is positive; P and N, the number of positive and negative examples.

Ouput: R, a list of ROC points increasing by false positive rate.

1. $L_{sorted} \leftarrow L$ **sorted decreasing by** f **scores**
2. $FP \leftarrow TP \leftarrow 0; R \leftarrow \langle\rangle; fprev \leftarrow -\infty; i \leftarrow 1$
3. **while** $i \leq |L_{sorted}|$
4. **if** $f(i) \neq f_{prev}$ **then**
5. $R.append\left(\frac{FP}{N}, \frac{TP}{P}\right)$
6. $fprev \leftarrow f(i)$
7. **if** $L_{sorted}(i)$ **is a positive example then** $TP = TP + 1$

We have

Algorithm ROC point generation

Input: L, the set of test examples; $f(i)$, the probabilistic classifier estimate that example i is positive; P and N, the number of positive and negative examples.

Ouput: R, a list of ROC points increasing by false positive rate.

1. $L_{sorted} \leftarrow L$ **sorted decreasing by** f **scores**
2. $FP \leftarrow TP \leftarrow 0; R \leftarrow \langle\rangle; fprev \leftarrow -\infty; i \leftarrow 1$
3. **while** $i \leq |L_{sorted}|$
4. **if** $f(i) \neq f_{prev}$ **then**
5. $R.append\left(\frac{FP}{N}, \frac{TP}{P}\right)$
6. $fprev \leftarrow f(i)$
7. **if** $L_{sorted}(i)$ **is a positive example then** $TP = TP + 1$
8. **else** $FP = FP + 1$

We have

Algorithm ROC point generation

Input: L, the set of test examples; $f(i)$, the probabilistic classifier estimate that example i is positive; P and N, the number of positive and negative examples.

Ouput: R, a list of ROC points increasing by false positive rate.

1. $L_{sorted} \leftarrow L$ **sorted decreasing by** f **scores**
2. $FP \leftarrow TP \leftarrow 0; R \leftarrow \langle\rangle; fprev \leftarrow -\infty; i \leftarrow 1$
3. **while** $i \leq |L_{sorted}|$
4. **if** $f(i) \neq f_{prev}$ **then**
5. $R.append\left(\frac{FP}{N}, \frac{TP}{P}\right)$
6. $fprev \leftarrow f(i)$
7. **if** $L_{sorted}(i)$ **is a positive example then** $TP = TP + 1$
8. **else** $FP = FP + 1$
9. $i \leftarrow i + 1$

We have

Algorithm ROC point generation

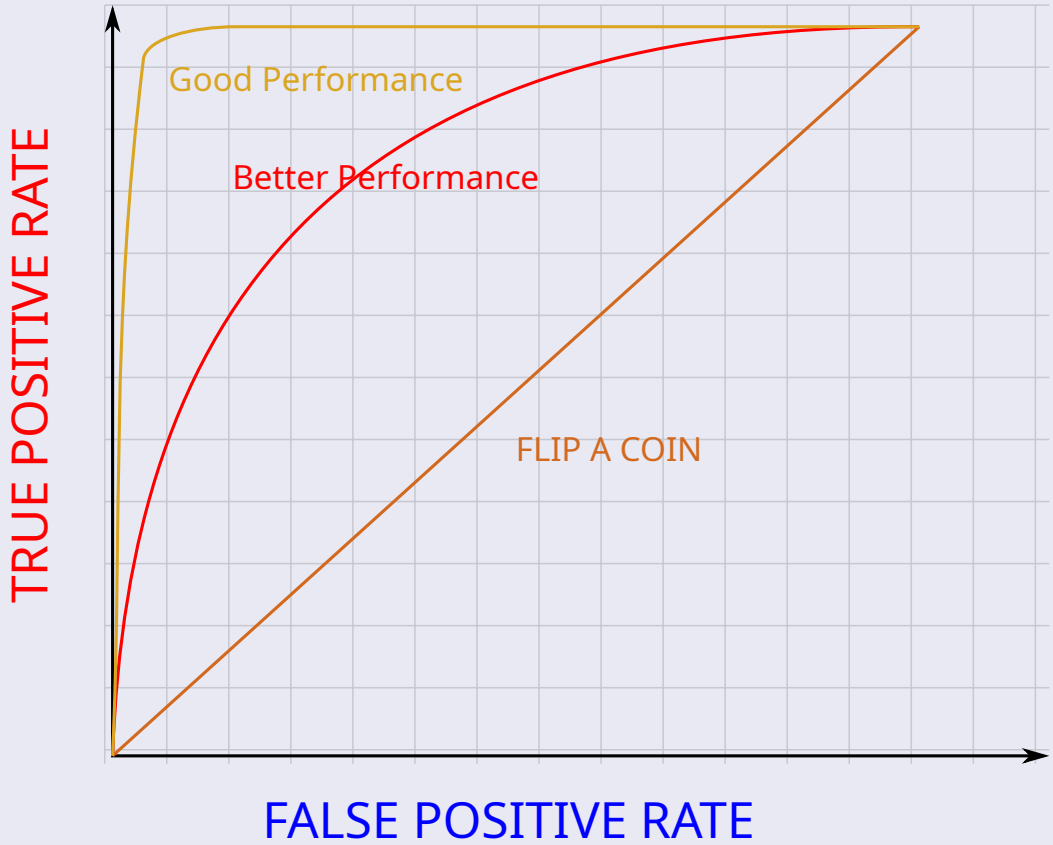
Input: L, the set of test examples; $f(i)$, the probabilistic classifier estimate that example i is positive; P and N, the number of positive and negative examples.

Ouput: R, a list of ROC points increasing by false positive rate.

1. $L_{sorted} \leftarrow L$ **sorted decreasing by** f **scores**
2. $FP \leftarrow TP \leftarrow 0; R \leftarrow \langle\rangle; fprev \leftarrow -\infty; i \leftarrow 1$
3. **while** $i \leq |L_{sorted}|$
4. **if** $f(i) \neq f_{prev}$ **then**
5. $R.append\left(\frac{FP}{N}, \frac{TP}{P}\right)$
6. $fprev \leftarrow f(i)$
7. **if** $L_{sorted}(i)$ **is a positive example then** $TP = TP + 1$
8. **else** $FP = FP + 1$
9. $i \leftarrow i + 1$
10. $R.append\left(\frac{FP}{N}, \frac{TP}{P}\right)$**, this is** $(1, 1)$

For Example

We could have multiple methods



Thus

Thus

- Thus, after generating the ROC Curve it is possible to use several metrics to validate using the ROC curves.

Thus

Thus

- Thus, after generating the ROC Curve it is possible to use several metrics to validate using the ROC curves.

A Partial List is

1. Area Under the Curve (AUC)
2. Equal Error Rate (EER)
3. Likelihood Ratio

Outline

1. Bias-Variance Dilemma
   - Introduction
   - Measuring the difference between optimal and learned
   - The Bias-Variance
   - "Extreme" Example
2. Confusion Matrix
   - Introduction
   - Statistical Testing
   - The α and β errors
   - The Initial Confusion Matrix
     - Metrics from the Confusion Matrix
   - The Multi Class Problem
   - Application in Computer Vision
   - Precision-Recall Curve
     - Application: Average Precision
   - Precision and Recall in the BERT Language Model
3. Receiver Operator Curves (ROC)
   - Introduction
   - Example
   - Algorithm for the ROC Curve
   - Area Under the Curve (AUC)
   - Other Measures: F_1-Measure
4. Cross Validation
   - Introduction
   - How to choose K
   - Types of Cross Validation
   - Solving The Imbalanced Class Problem

A Simple Defintion

We have

$$AUC = \int ROC\left(p\right) dp = \sum_{i=1}^{N} ROC\left(f\left(\frac{1}{i}\right)\right)\left[\frac{i}{N} - \frac{i-1}{N}\right]$$

A Simple Defintion

We have

$$AUC = \int ROC\left(p\right) dp = \sum_{i=1}^{N} ROC\left(f\left(\frac{1}{i}\right)\right)\left[\frac{i}{N} - \frac{i-1}{N}\right]$$

This equation has the following meaning

- The probability that a randomly selected observation X from the **positive class** would have a higher score than a randomly selected observation Y from the **negative class.**

$$P\left(X > Y\right)$$

A Simple Defintion

We have

$$AUC = \int ROC\left(p\right) dp = \sum_{i=1}^{N} ROC\left(f\left(\frac{1}{i}\right)\right)\left[\frac{i}{N} - \frac{i-1}{N}\right]$$

This equation has the following meaning

- The probability that a randomly selected observation X from the **positive class** would have a higher score than a randomly selected observation Y from the **negative class.**

$$P\left(X > Y\right)$$

Thus

The AUC gives the mean **true positive** rate averaged uniformly across the **false positive** rate.

Therefore

AUC curves are a good measure of how good are our results

- However, we need to combine this results with something more powerful
  - Cross Validation - to understand the variation in the machine estimation

Outline

1. Bias-Variance Dilemma
   - Introduction
   - Measuring the difference between optimal and learned
   - The Bias-Variance
   - "Extreme" Example
2. Confusion Matrix
   - Introduction
   - Statistical Testing
   - The α and β errors
   - The Initial Confusion Matrix
     - Metrics from the Confusion Matrix
   - The Multi Class Problem
   - Application in Computer Vision
   - Precision-Recall Curve
     - Application: Average Precision
   - Precision and Recall in the BERT Language Model
3. Receiver Operator Curves (ROC)
   - Introduction
   - Example
   - Algorithm for the ROC Curve
   - Area Under the Curve (AUC)
   - Other Measures: F_1-Measure
4. Cross Validation
   - Introduction
   - How to choose K
   - Types of Cross Validation
   - Solving The Imbalanced Class Problem

Also known as F_1 score

It is a measure of a test's accuracy

- It considers both the precision P and the recall R of the test to compute the score.

Also known as F_1 score

It is a measure of a test's accuracy

- It considers both the precision P and the recall R of the test to compute the score.

An interesting fact

- It computes some average of the information retrieval precision and recall.

Remember

Precision

- The proportion of the elements classified as true positive vs the total of all the real true positives.

$$\text{Precision Predicted Value} = \frac{TP}{FP + TP}$$

Remember

Precision

- The proportion of the elements classified as true positive vs the total of all the real true positives.

$$\text{Precision Predicted Value} = \frac{TP}{FP + TP}$$

Recall

- True Positive Rate is the proportion of getting a correct classification of the Positive Class vs the True Positive and False Negatives.

$$\text{True Positive Rate} = \frac{TP}{TP + FN}$$

Building the F_1 score

Something Notable

$$Average = \frac{1}{N}\sum_{i=1}^{N} x_i$$

$$Harmonic = \frac{N}{\sum_{i=1}^{N} \frac{1}{x_i}}$$

Building the F_1 score

Something Notable

$$Average = \frac{1}{N}\sum_{i=1}^{N} x_i$$

$$Harmonic = \frac{N}{\sum_{i=1}^{N} \frac{1}{x_i}}$$

When $x_1 = Precision$ and $x_2 = Recall$

$$Average = \frac{1}{2}(P + R)$$

$$Harmonic = \frac{2}{\frac{1}{P} + \frac{1}{R}} = \frac{2PR}{P + R}$$

Thus

Important

- The harmonic mean is more intuitive than the arithmetic mean when computing a mean of ratios.

Thus

Important

- The harmonic mean is more intuitive than the arithmetic mean when computing a mean of ratios.

Example

- Suppose that you have a finger print recognition system and its precision and recall be 1.0 and 0.2

Therefore, we have

The Average will report

$$\frac{1}{2}(P+R) = \frac{1.0+0.2}{2} = 0.6$$

Therefore, we have

The Average will report

$$\frac{1}{2}(P+R) = \frac{1.0+0.2}{2} = 0.6$$

At the F_2 score

$$\frac{2PR}{P+R} = \frac{0.4}{1.2} = 0.33$$

General Form F_β

Then for Precision and Recall, we have a general function

$$F_\beta = \frac{(\beta^2 + 1)\, Precision \times Recall}{\beta^2 Precision + Recall} \quad (0 \leq \beta \leq +\infty)$$

General Form F_β

Then for Precision and Recall, we have a general function

$$F_\beta = \frac{(\beta^2 + 1)\, Precision \times Recall}{\beta^2 Precision + Recall} \quad (0 \leq \beta \leq +\infty)$$

Thus, for the basic case F_1

$$F_1 = 2\frac{Precision \times Recall}{Precision + Recall}$$

Outline

1. Bias-Variance Dilemma
   - Introduction
   - Measuring the difference between optimal and learned
   - The Bias-Variance
   - "Extreme" Example
2. Confusion Matrix
   - Introduction
   - Statistical Testing
   - The α and β errors
   - The Initial Confusion Matrix
     - Metrics from the Confusion Matrix
   - The Multi Class Problem
   - Application in Computer Vision
   - Precision-Recall Curve
     - Application: Average Precision
   - Precision and Recall in the BERT Language Model
3. Receiver Operator Curves (ROC)
   - Introduction
   - Example
   - Algorithm for the ROC Curve
   - Area Under the Curve (AUC)
   - Other Measures: F_1-Measure
4. Cross Validation
   - Introduction
   - How to choose K
   - Types of Cross Validation
   - Solving The Imbalanced Class Problem

What we want

We want to measure

A quality measure to measure different classifiers (for different parameter values).

What we want

We want to measure

A quality measure to measure different classifiers (for different parameter values).

We call that as

$$R(f) = E_{\mathcal{D}}\left[L\left(y, f\left(\boldsymbol{x}\right)\right)\right]. \tag{8}$$

Example: $L\left(y, f\left(x\right)\right) = \left\|y - f\left(\boldsymbol{x}\right)\right\|_2^2$

What we want

We want to measure

A quality measure to measure different classifiers (for different parameter values).

We call that as

$$R(f) = E_{\mathcal{D}}\left[L\left(y, f\left(\boldsymbol{x}\right)\right)\right]. \tag{8}$$

Example: $L\left(y, f\left(x\right)\right) = \left\|y - f\left(\boldsymbol{x}\right)\right\|_2^2$

More precisely

- For different values γ_j of the parameter, we train a classifier $f\left(\boldsymbol{x}|\gamma_j\right)$ on the training set.

Then, calculate the empirical Risk

Do you have any ideas?

Give me your best shot!!!

Then, calculate the empirical Risk

Do you have any ideas?

Give me your best shot!!!

Empirical Risk

We use the validation set to estimate

$$\hat{R}\left(f\left(x|\gamma\right)\right) = \frac{1}{N_v}\sum_{i=1}^{N_v} L\left(y_i, f\left(\boldsymbol{x}_i|\gamma\right)\right) \tag{9}$$

Then, calculate the empirical Risk

Do you have any ideas?

Give me your best shot!!!

Empirical Risk

We use the validation set to estimate

$$\hat{R}\left(f\left(x|\gamma\right)\right)=\frac{1}{N_v}\sum_{i=1}^{N_v}L\left(y_i,f\left(\boldsymbol{x}_i|\gamma\right)\right) \tag{9}$$

Thus, you follow the following procedure

1. Select the value γ^* which achieves the smallest estimated error.

Then, calculate the empirical Risk

Do you have any ideas?

Give me your best shot!!!

Empirical Risk

We use the validation set to estimate

$$\hat{R}\left(f\left(x|\gamma\right)\right) = \frac{1}{N_v}\sum_{i=1}^{N_v} L\left(y_i, f\left(\boldsymbol{x}_i|\gamma\right)\right) \tag{9}$$

Thus, you follow the following procedure

1. Select the value γ^* which achieves the smallest estimated error.
2. Re-train the classifier with parameter γ^* on all data except the test set (i.e. train + validation data).

Then, calculate the empirical Risk

Do you have any ideas?

Give me your best shot!!!

Empirical Risk

We use the validation set to estimate

$$\hat{R}\left(f\left(x|\gamma\right)\right) = \frac{1}{N_v}\sum_{i=1}^{N_v} L\left(y_i, f\left(\boldsymbol{x}_i|\gamma\right)\right) \tag{9}$$

Thus, you follow the following procedure

1. Select the value γ^* which achieves the smallest estimated error.
2. Re-train the classifier with parameter γ^* on all data except the test set (i.e. train + validation data).
3. Report error estimate $\hat{R}\left(f\left(x|\gamma^*\right)\right)$ computed on the test set.

Idea

We want to have

- An estimation that allows us to see how to test all the data in a fair way assuming: Train, Validation and Test scenarios

Idea

We want to have

- An estimation that allows us to see how to test all the data in a fair way assuming: Train, Validation and Test scenarios

K-fold Cross Validation

To estimate the risk of a classifier f:

1. Split data into K equally sized parts (called "folds"), N_v.
2. Train an instance f_k of the classifier, using all folds except fold k as training data.

Idea

We want to have

- An estimation that allows us to see how to test all the data in a fair way assuming: Train, Validation and Test scenarios

K-fold Cross Validation

To estimate the risk of a classifier f:

1. Split data into K equally sized parts (called "folds"), N_v.
2. Train an instance f_k of the classifier, using all folds except fold k as training data.
3. Compute the Cross Validation (CV) estimate:

$$\hat{R}_{CV}\left(f\left(x|\gamma\right)\right) = \frac{1}{N}\sum_{k=1}^{K}\sum_{i=1}^{N} L\left(y_i, f_k\left(\boldsymbol{x}_{k(i)}|\gamma\right)\right) \tag{10}$$

where $k\left(i\right)$ is the fold containing $\boldsymbol{x}_i$.

Example

$K = 5, k = 3$

Train	Train	Testing	Train	Train
1	2	3	4	5

Example

$K = 5, k = 3$

Train	Train	Testing	Train	Train
1	2	3	4	5

Actually, we have

- A more general setup

$\overbrace{\text{SPLIT All Train Set}}^{\text{Train Data + Validation Data}}$	Test

Outline

1. Bias-Variance Dilemma
   - Introduction
   - Measuring the difference between optimal and learned
   - The Bias-Variance
   - "Extreme" Example
2. Confusion Matrix
   - Introduction
   - Statistical Testing
   - The α and β errors
   - The Initial Confusion Matrix
     - Metrics from the Confusion Matrix
   - The Multi Class Problem
   - Application in Computer Vision
   - Precision-Recall Curve
     - Application: Average Precision
   - Precision and Recall in the BERT Language Model
3. Receiver Operator Curves (ROC)
   - Introduction
   - Example
   - Algorithm for the ROC Curve
   - Area Under the Curve (AUC)
   - Other Measures: F_1-Measure
4. Cross Validation
   - Introduction
   - How to choose K
   - Types of Cross Validation
   - Solving The Imbalanced Class Problem

How to choose K

Extremal cases

- $K = N$, called leave one out cross validation (loocv)

How to choose K

Extremal cases

- $K = N$, called leave one out cross validation (loocv)
- $K = 2$

How to choose K

Extremal cases

- $K = N$, called leave one out cross validation (loocv)
- $K = 2$

An often cited problem with loocv is that we have to train many ($= N$) classifiers, but there is also a deeper problem.

How to choose K

Extremal cases

- $K = N$, called leave one out cross validation (loocv)
- $K = 2$

An often cited problem with loocv is that we have to train many ($= N$) classifiers, but there is also a deeper problem.

Argument 1: K should be small, e.g. $K = 2$

1. Unless we have a lot of data, variance between two distinct training sets may be considerable.

How to choose K

Extremal cases

- $K = N$, called leave one out cross validation (loocv)
- $K = 2$

An often cited problem with loocv is that we have to train many ($= N$) classifiers, but there is also a deeper problem.

Argument 1: K should be small, e.g. $K = 2$

1. Unless we have a lot of data, variance between two distinct training sets may be considerable.
2. Important concept: By removing substantial parts of the sample in turn and at random, we can simulate this variance.

How to choose K

Extremal cases

- $K = N$, called leave one out cross validation (loocv)
- $K = 2$

An often cited problem with loocv is that we have to train many ($= N$) classifiers, but there is also a deeper problem.

Argument 1: K should be small, e.g. $K = 2$

1. Unless we have a lot of data, variance between two distinct training sets may be considerable.
2. Important concept: By removing substantial parts of the sample in turn and at random, we can simulate this variance.
3. By removing a single point (loocv), we cannot make this variance visible.

How to choose K

Argument 2: K should be large, e.g, $K = 10$

1. Classifiers generally perform better when trained on larger data sets.

How to choose K

Argument 2: K should be large, e.g, $K = 10$

1. Classifiers generally perform better when trained on larger data sets.
2. What a small K means? We substantially reduce the amount of training data used to train each f_k, so we may end up with weaker classifiers.

How to choose K

Argument 2: K should be large, e.g, $K = 10$

1. Classifiers generally perform better when trained on larger data sets.
2. What a small K means? We substantially reduce the amount of training data used to train each f_k, so we may end up with weaker classifiers.
3. This way, we will systematically overestimate the risk because .

How to choose K

Argument 2: K should be large, e.g, $K = 10$

1. Classifiers generally perform better when trained on larger data sets.
2. What a small K means? We substantially reduce the amount of training data used to train each f_k, so we may end up with weaker classifiers.
3. This way, we will systematically overestimate the risk because .

Common recommendation: $K = 5$ to $K = 10$

Intuition:

How to choose K

Argument 2: K should be large, e.g, $K = 10$

1. Classifiers generally perform better when trained on larger data sets.
2. What a small K means? We substantially reduce the amount of training data used to train each f_k, so we may end up with weaker classifiers.
3. This way, we will systematically overestimate the risk because .

Common recommendation: $K = 5$ to $K = 10$

Intuition:

1. $K = 10$ means number of samples removed from training is one order of magnitude below training sample size.

How to choose K

Argument 2: K should be large, e.g, $K = 10$

1. Classifiers generally perform better when trained on larger data sets.
2. What a small K means? We substantially reduce the amount of training data used to train each f_k, so we may end up with weaker classifiers.
3. This way, we will systematically overestimate the risk because .

Common recommendation: $K = 5$ to $K = 10$

Intuition:

1. $K = 10$ means number of samples removed from training is one order of magnitude below training sample size.
2. This should not weaken the classifier considerably, but should be large enough to make measure variance effects.

Outline

1. Bias-Variance Dilemma
   - Introduction
   - Measuring the difference between optimal and learned
   - The Bias-Variance
   - "Extreme" Example
2. Confusion Matrix
   - Introduction
   - Statistical Testing
   - The α and β errors
   - The Initial Confusion Matrix
     - Metrics from the Confusion Matrix
   - The Multi Class Problem
   - Application in Computer Vision
   - Precision-Recall Curve
     - Application: Average Precision
   - Precision and Recall in the BERT Language Model
3. Receiver Operator Curves (ROC)
   - Introduction
   - Example
   - Algorithm for the ROC Curve
   - Area Under the Curve (AUC)
   - Other Measures: F_1-Measure
4. Cross Validation
   - Introduction
   - How to choose K
   - Types of Cross Validation
   - Solving The Imbalanced Class Problem

Leave p out cross-validation

Definition

- It involves using p-observation as test data, and remaining data is used to train the model.

Leave p out cross-validation

Definition

- It involves using p-observation as test data, and remaining data is used to train the model.

Basically

- This is repeated in all ways to cut the original sample on a test set of p observations and a training set.

Leave p out cross-validation

Definition

- It involves using p-observation as test data, and remaining data is used to train the model.

Basically

- This is repeated in all ways to cut the original sample on a test set of p observations and a training set.

Notes

- A variant of LpOCV with $p = 2$ known as leave-pair-out cross-validation has been recommended as a nearly unbiased method for estimating the area under ROC curve of a binary classifier.

Leave-one-out cross-validation (LOOCV)

Definition

- It is a category of LpOCV with the case of $p = 1$.

Leave-one-out cross-validation (LOOCV)

Definition

- It is a category of LpOCV with the case of $p = 1$.

Basically

Train	Train	Train	Test $= 1$	Train	Train

Pros and Cons

Pros

- Simple, easy to understand, and implement.

Pros and Cons

Pros

- Simple, easy to understand, and implement.

Cons

- The model may lead to a low bias.
- The computation time required is high.

Holdout Cross Validation

Definition

- The holdout technique is an exhaustive Cross Validation method.
- It randomly splits the dataset into train and test data.
  - For example, 70% for Train+Validation and 30% for Test

Holdout Cross Validation

Definition

- The holdout technique is an exhaustive Cross Validation method.
- It randomly splits the dataset into train and test data.
  - For example, 70% for Train+Validation and 30% for Test

Pros

- Simple to understand

Holdout Cross Validation

Definition

- The holdout technique is an exhaustive Cross Validation method.
- It randomly splits the dataset into train and test data.
  - For example, 70% for Train+Validation and 30% for Test

Pros

- Simple to understand

Cons

- Not suitable for an imbalanced dataset.
- Requires large amount of data

K-Fold Cross Validation

Definition

- In k-fold Cross Validation, the original dataset is equally partitioned into k sub-parts or folds.

K-Fold Cross Validation

Definition

- In k-fold Cross Validation, the original dataset is equally partitioned into k sub-parts or folds.

Thus

- Out of the k-folds or groups, for each iteration, one group is selected as test data,
- The remaining $(k-1)$ groups are selected as training data for the train+validation scenario

With this K Cross Validation

We take the mean accuracy of the k-folds

$$acc_{cv} = \frac{1}{K}\sum_{i=1}^{K} acc_i$$

With this K Cross Validation

We take the mean accuracy of the k-folds

$$acc_{cv} = \frac{1}{K} \sum_{i=1}^{K} acc_i$$

Pros

- The model has low bias and Low time complexity
- The entire dataset is utilized for both training and validation.

With this K Cross Validation

We take the mean accuracy of the k-folds

$$acc_{cv} = \frac{1}{K} \sum_{i=1}^{K} acc_i$$

Pros

- The model has low bias and Low time complexity
- The entire dataset is utilized for both training and validation.

Cons

- Not suitable for an imbalanced dataset.

Repeated Sub-sampling

Definition

- Repeated random sub-sampling validation also referred to as Monte Carlo cross-validation splits the dataset randomly into training+validation and testing.

Repeated Sub-sampling

Definition

- Repeated random sub-sampling validation also referred to as Monte Carlo cross-validation splits the dataset randomly into training+validation and testing.

Something Notable

- Unlikely k-fold cross-validation, k-folds, the splits are done randomly.
- Therefore the need of using multiple iterations to perform an average accuracy

Finally

Pros

- The proportion of train+validation and test splits is not dependent on the number of iterations or partitions.

Finally

Pros

- The proportion of train+validation and test splits is not dependent on the number of iterations or partitions.

Cons

- Some samples may not be selected for either training or validation.
- Not suitable for an imbalanced dataset.

Outline

1. Bias-Variance Dilemma
   - Introduction
   - Measuring the difference between optimal and learned
   - The Bias-Variance
   - "Extreme" Example
2. Confusion Matrix
   - Introduction
   - Statistical Testing
   - The α and β errors
   - The Initial Confusion Matrix
     - Metrics from the Confusion Matrix
   - The Multi Class Problem
   - Application in Computer Vision
   - Precision-Recall Curve
     - Application: Average Precision
   - Precision and Recall in the BERT Language Model
3. Receiver Operator Curves (ROC)
   - Introduction
   - Example
   - Algorithm for the ROC Curve
   - Area Under the Curve (AUC)
   - Other Measures: F_1-Measure
4. Cross Validation
   - Introduction
   - How to choose K
   - Types of Cross Validation
   - Solving The Imbalanced Class Problem

What is stratified sampling?

Before we talk about the solution

- What is stratified sampling?

What is stratified sampling?

Before we talk about the solution

- What is stratified sampling?

Stratified sampling is a sampling technique where the samples are selected in the same proportion

- by dividing the population into groups called ‘strata’ based on a characteristic

For Example, male and female population

For example, if the population of interest has 30% male and 70% female subjects

- Instead of randomly sampling the entire population, we generate a strata

For Example, male and female population

For example, if the population of interest has 30% male and 70% female subjects

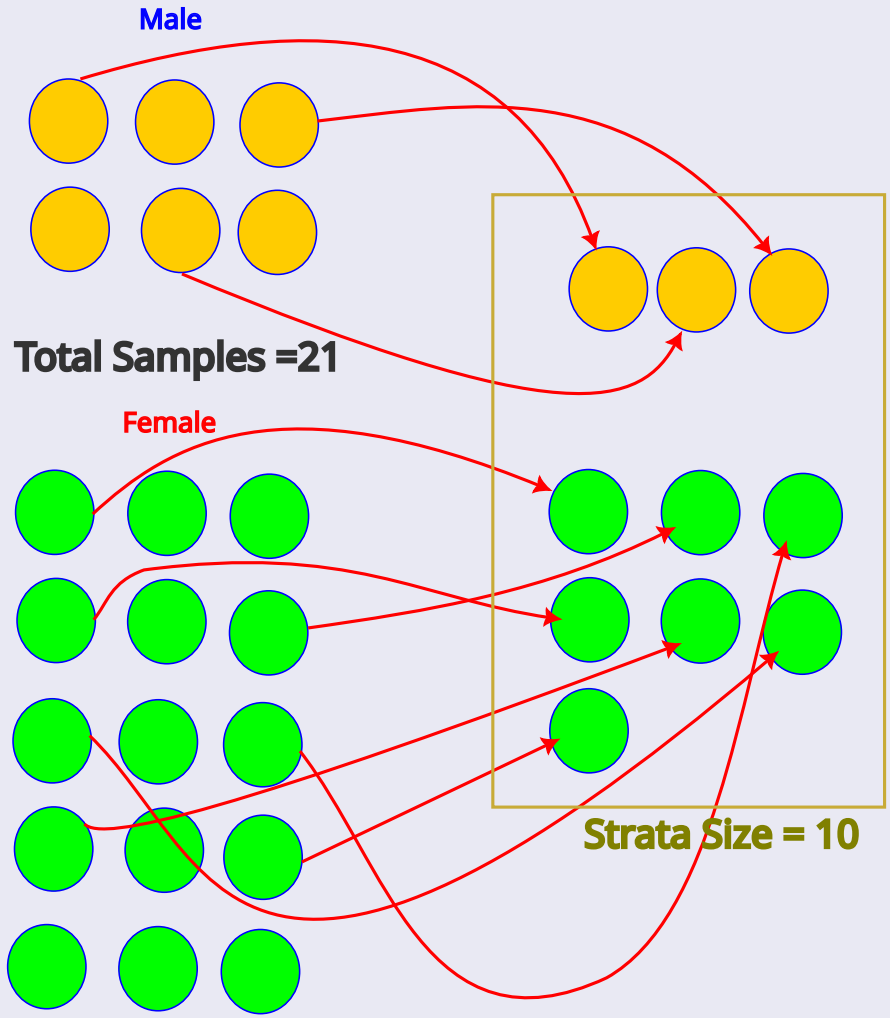
- Instead of randomly sampling the entire population, we generate a strata

In our example

- We sample 30% from the male population 70% of the female population

Or as the following figure shows

We have by randomly sampling each percentage



Stratified K-Fold Cross-Validation

We have the following situation

- For all the cross-validation techniques discussed above, they may not work well with an imbalanced dataset.
  - Stratified k-fold cross-validation tries to solve the problem of an imbalanced dataset.

Stratified K-Fold Cross-Validation

We have the following situation

- For all the cross-validation techniques discussed above, they may not work well with an imbalanced dataset.
  - Stratified k-fold cross-validation tries to solve the problem of an imbalanced dataset.

Definition

- In Stratified k-fold cross-validation, the dataset is partitioned into k groups or folds
  - the training and test sets have the same proportion of the feature of interest as in the original dataset.
  - As in stratified sampling

Therefore

Final Score

- The final score is computed by taking the mean of scores of each fold.

Therefore

Final Score

- The final score is computed by taking the mean of scores of each fold.

Pros

- Works well for an imbalanced dataset.

Therefore

Final Score

- The final score is computed by taking the mean of scores of each fold.

Pros

- Works well for an imbalanced dataset.

Cons

- Now suitable for time series dataset.

Therefore

We can add an extra for fine tuning hyper parameters

- We split the training and validation in same size samples to find the best hyper parameters

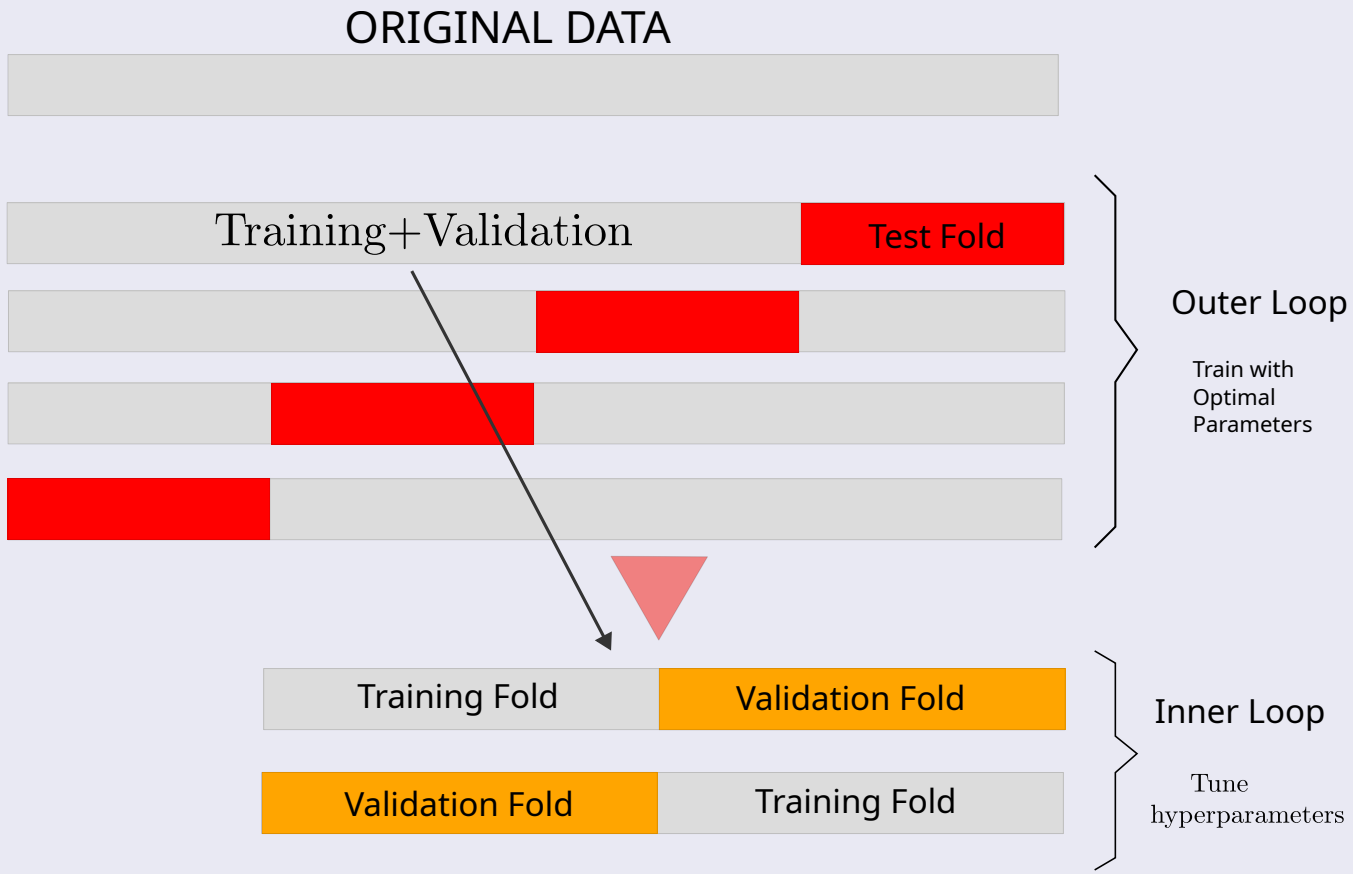
Therefore

We can add an extra for fine tuning hyper parameters

- We split the training and validation in same size samples to find the best hyper parameters

Or any crazy partition

- To get the optimal hyper parameters.

We have then

When splitting Training+Validation in equal size sets