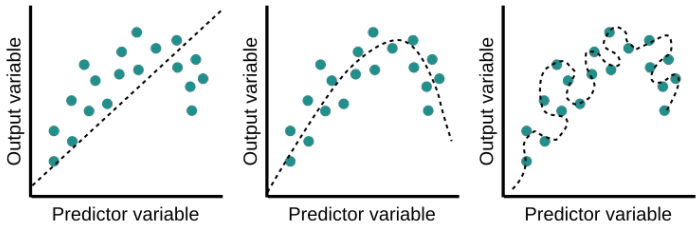
Overfitting and Underfitting

Tushar B. Kute,
http://tusharkute.com



Lets have an example

- Even when we're working on a machine learning project, we often face situations where we are encountering unexpected performance or error rate differences between the training set and the test set (as shown below).
- How can a model perform so well over the training set and just as poorly on the test set?

Reference: Analytics Vidhya

Example

```
1 from sklearn.metrics import classification_report
2 print(classification_report(y_train, predicted_values))
```

	precision	recall	f1-score	support
0	1.00	1.00	1.00	14234
1	1.00	1.00	1.00	3419
accuracy			1.00	17653
macro avg	1.00	1.00	1.00	17653
weighted avg	1.00	1.00	1.00	17653

```
1 predicted_values = classifier.predict(x_test)
2 print(classification_report(y_test, predicted_values))
```

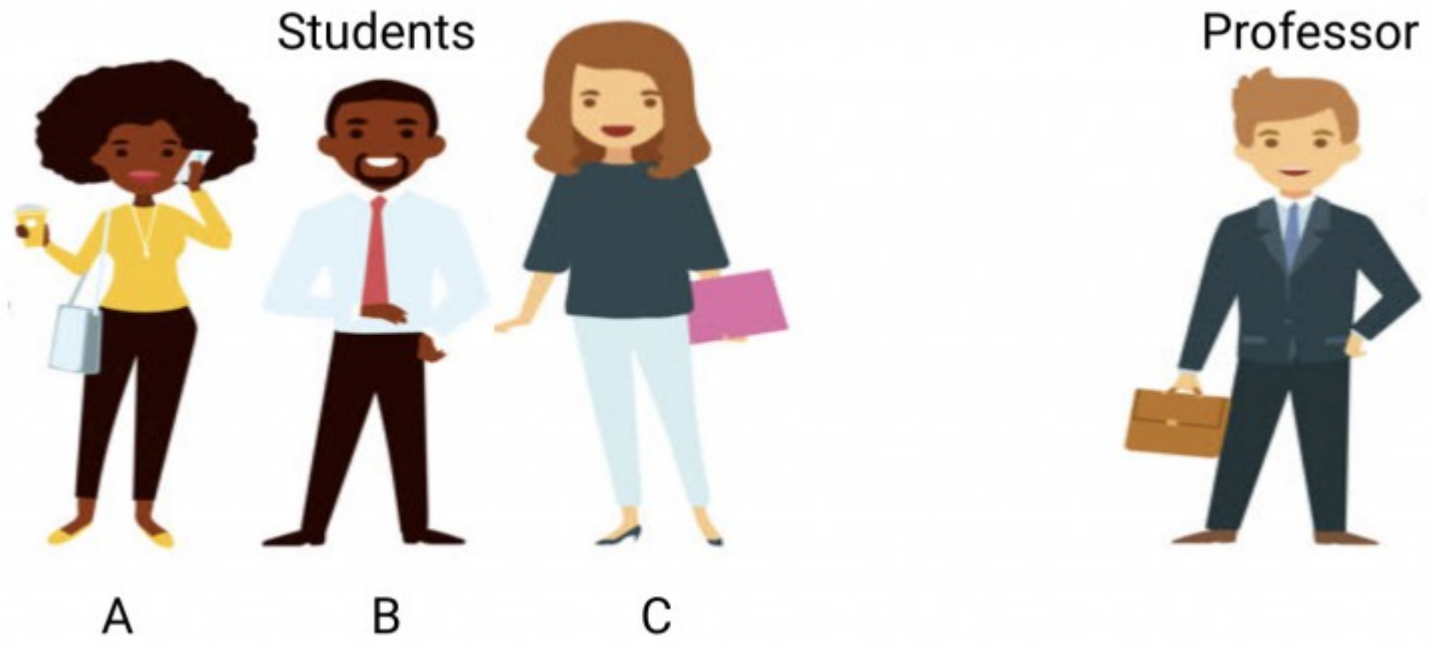
	precision	recall	f1-score	support
0	0.87	0.86	0.87	3559
1	0.44	0.46	0.45	855
accuracy			0.78	4414
macro avg	0.66	0.66	0.66	4414
weighted avg	0.79	0.78	0.79	4414

Example

- This happens very frequently whenever I am working with tree-based predictive models. Because of the way the algorithms work, you can imagine how tricky it is to avoid falling into the overfitting trap!
- Moreover, it can be quite daunting when we are unable to find the underlying reason why our predictive model is exhibiting this anomalous behavior.
- Here's my personal experience – ask any seasoned data scientist about this, they typically start talking about some array of fancy terms like Overfitting, Underfitting, Bias, and Variance. But little does anyone talk about the intuition behind these machine learning concepts.

Example

- Consider a math class consisting of 3 students and a professor.



Example

- Now, in any classroom, we can broadly divide the students into 3 categories. We'll talk about them one-by-one.

A

- Hobby = chating
- Not interested in class
- Doesn't pay much attention to professor

- Let's say that student A resembles a student who does not like math. She is not interested in what is being taught in the class and therefore does not pay much attention to the professor and the content he is teaching.

Example

- Let’s consider student B. He is the most competitive student who focuses on memorizing each and every question being taught in class instead of focusing on the key concepts. Basically, he isn’t interested in learning the problem-solving approach.

B

- Hobby = to be best in class.
- Mugs up everything professor says.
- Too much attention to the class work.

Example

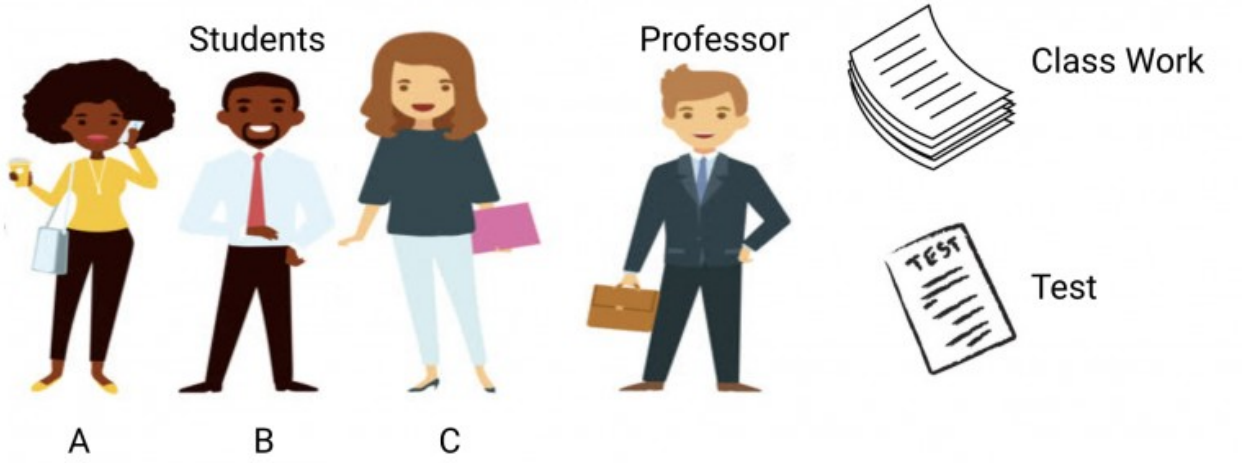
- Finally, we have the ideal student C. She is purely interested in learning the key concepts and the problem-solving approach in the math class rather than just memorizing the solutions presented.

C

- Hobby = learning new things
- Eager to learn concepts.
- Pays attention to class and learns the idea behind solving a problem.

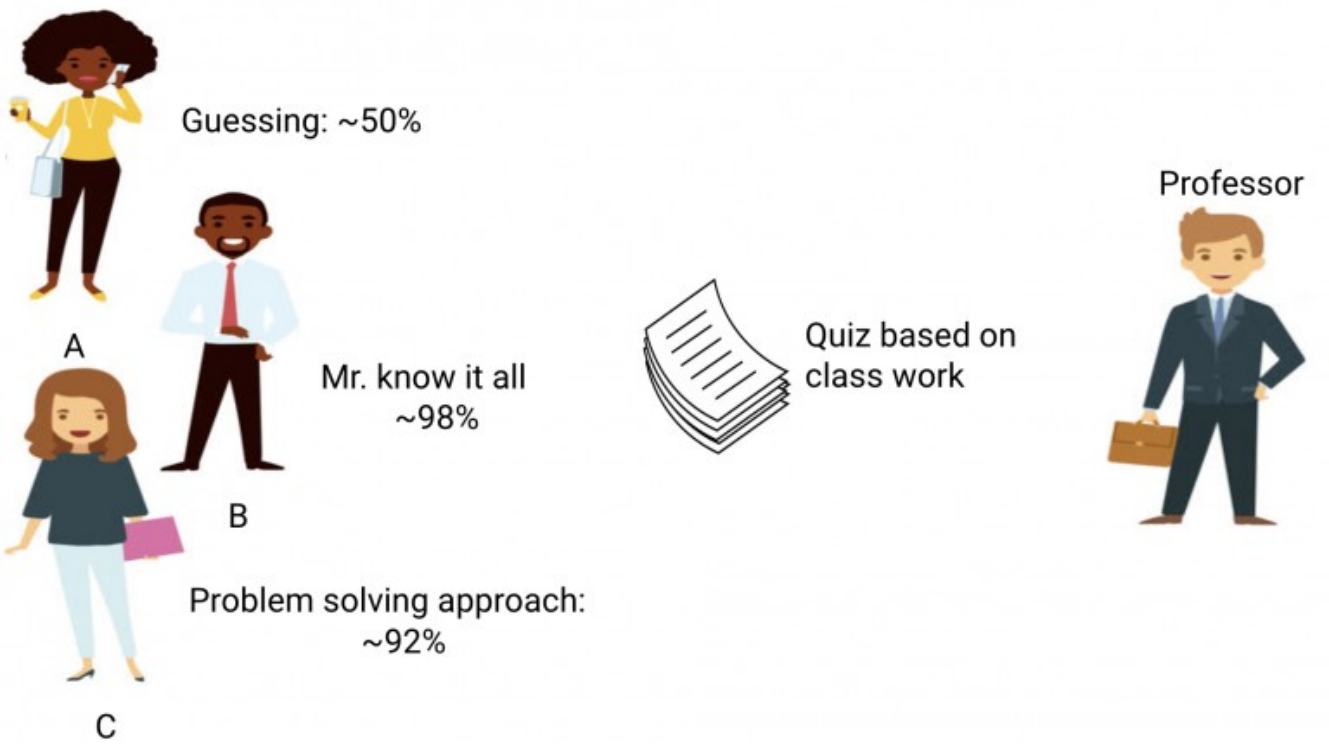
Example

- We all know from experience what happens in a classroom. The professor first delivers lectures and teaches the students about the problems and how to solve them. At the end of the day, the professor simply takes a quiz based on what he taught in the class.



Example

- So, let’s discuss what happens when the teacher takes a classroom test at the end of the day:

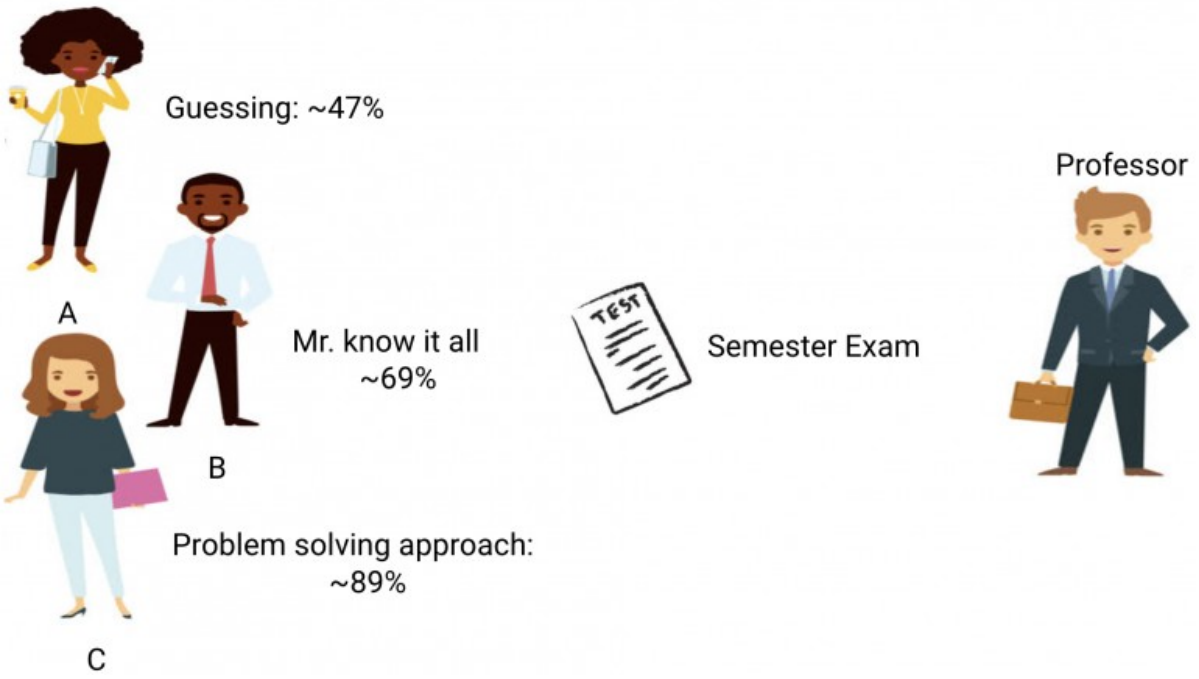


Example

- Student A, who was distracted in his own world, simply guessed the answers and got approximately 50% marks in the test
- On the other hand, the student who memorized each and every question taught in the classroom was able to answer almost every question by memory and therefore obtained 98% marks in the class test
- For student C, she actually solved all the questions using the problem-solving approach she learned in the classroom and scored 92%

Example

- Now here’s the twist. Let’s also look at what happens during the monthly test, when students have to face new unknown questions which are not taught in the class by the teacher.



Example

- In the case of student A, things did not change much and he still randomly answers questions correctly ~50% of the time.
- In the case of Student B, his score dropped significantly. Can you guess why? This is because he always memorized the problems that were taught in the class but this monthly test contained questions which he has never seen before. Therefore, his performance went down significantly
- In the case of Student C, the score remained more or less the same. This is because she focused on learning the problem-solving approach and therefore was able to apply the concepts she learned to solve the unknown questions.

Example

A

Not interested in learning

Class test ~50%
Test ~47%

B

Memorizing the lessons

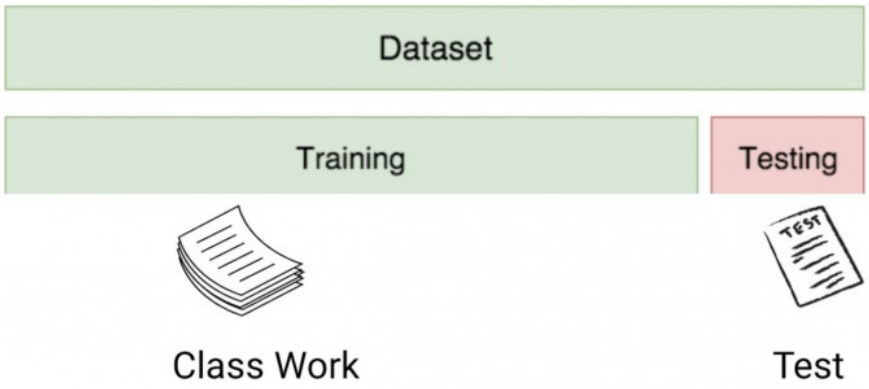
Class test ~98%
Test ~69%

C

Conceptual Learning

Class test ~92%
Test ~89%

Example



Now, recall our decision tree classifier I mentioned earlier. It gave a perfect score over the training set but struggled with the test set. Comparing that to the student examples we just discussed, the classifier establishes an analogy with student B who tried to memorize each and every question in the training set.

Conclusion

A

Not interested in learning

Class test ~50%
Test ~47%

Under-fit/ biased learning

B

Memorizing the lessons

Class test ~98%
Test ~69%

Over-fit/ Memorizing

C

Conceptual Learning

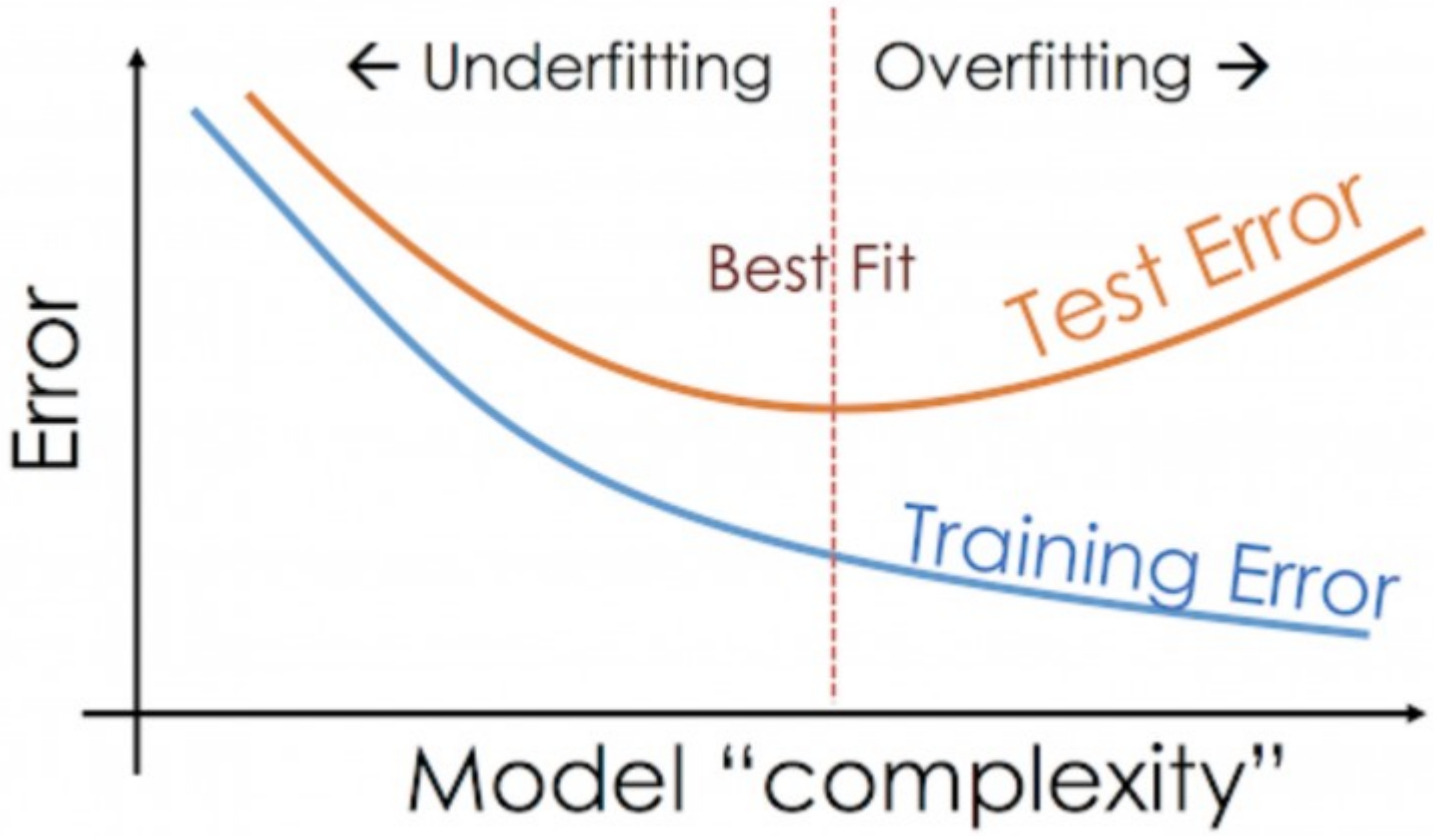
Class test ~92%
Test ~89%

Best-fit

Concept

- This situation where any given model is performing too well on the training data but the performance drops significantly over the test set is called an overfitting model.
- For example, non-parametric models like decision trees, KNN, and other tree-based algorithms are very prone to overfitting. These models can learn very complex relations which can result in overfitting.

Concept


← Underfitting
Overfitting →
Best Fit
Test Error
Training Error
Error
Model "complexity"

Concept

- On the other hand, if the model is performing poorly over the test and the train set, then we call that an underfitting model.
- An example of this situation would be building a linear regression model over non-linear data.

Overfitting

- Overfitting refers to a model that models the training data too well.
- Overfitting happens when a model learns the detail and noise in the training data to the extent that it negatively impacts the performance of the model on new data.
- This means that the noise or random fluctuations in the training data is picked up and learned as concepts by the model.
- The problem is that these concepts do not apply to new data and negatively impact the models ability to generalize.

Underfitting

- Underfitting refers to a model that can neither model the training data nor generalize to new data.
- An underfit machine learning model is not a suitable model and will be obvious as it will have poor performance on the training data.
- Underfitting is often not discussed as it is easy to detect given a good performance metric. The remedy is to move on and try alternate machine learning algorithms.
- Nevertheless, it does provide a good contrast to the problem of overfitting.

A good fit

- Ideally, you want to select a model at the sweet spot between underfitting and overfitting.
- This is the goal, but is very difficult to do in practice.
- To understand this goal, we can look at the performance of a machine learning algorithm over time as it is learning a training data.
- We can plot both the skill on the training data and the skill on a test dataset we have held back from the training process.

Limiting overfitting

- There are two important techniques that you can use when evaluating machine learning algorithms to limit overfitting:
  - Use a resampling technique to estimate model accuracy.
  - Hold back a validation dataset.

Summary

- Overfitting: Good performance on the training data, poor generliazation to other data.
- Underfitting: Poor performance on the training data and poor generalization to other data

Thank you

@mitu_skillologies

/mITuSkillologies

@mitu_group

/company/mitu-skillologies

MITUSkillologies

Web Resources
https://mitu.co.in
http://tusharkute.com

contact@mitu.co.in

tushar@tusharkute.com