Machine Learning

Machine Learning

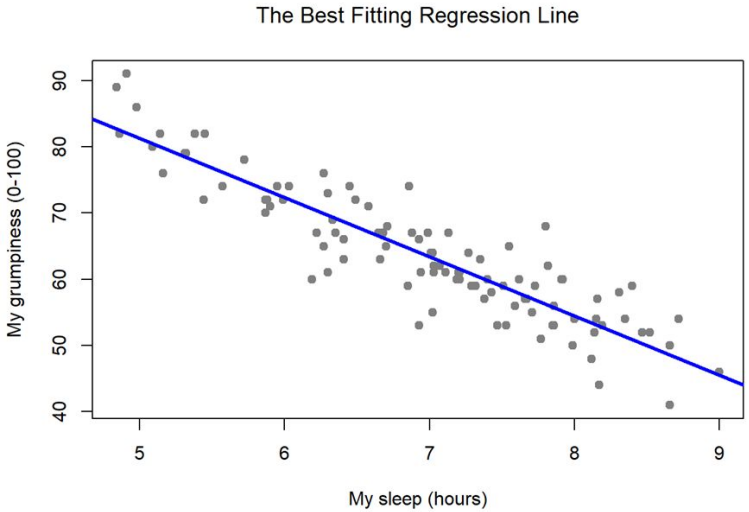
- Artificial intelligence is the ability of the computer to perform tasks that require human intelligence.
- Machine learning is a branch of artificial intelligence, it is the capability of the machine to imitate the way how human learn.
- This is achieved by using algorithm and models to draw inference from patterns in data.
- It is the process of building algorithms that can learn from the existing observations and predict new observations or determine the output of a new input.

Types of ML

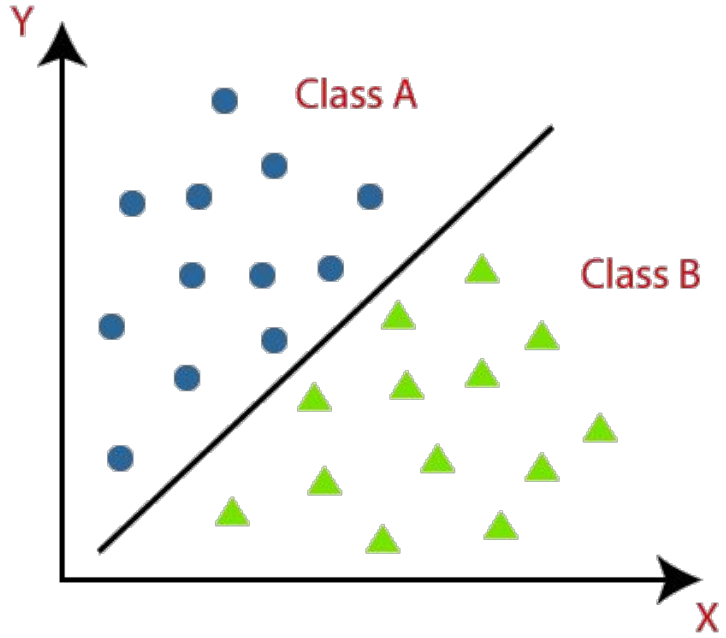
- Supervised learning: When the data sets are labeled.
  - Regression: The model finds the relationship between the output and one or more inputs. It predicts the output based on that relationship.
  - Classification: The model can identify the category of the new observation based on features or characteristics.
- Unsupervised learning: The datasets are not labeled.
- Reinforcement learning: Feedback based process and the model learns from its experience.

Supervised Learning

Regression and Classification algorithms



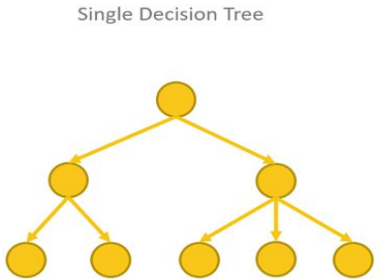
Regression

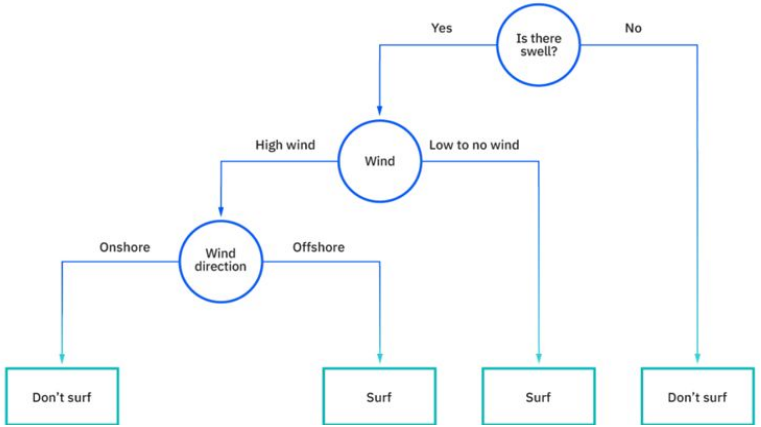


Classification

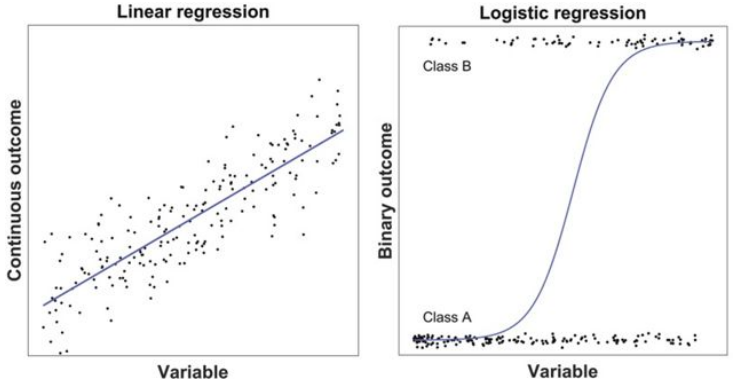
Classical ML Algorithms

- Decision tree:
  - Decision trees are used for prediction and classification.
  - It consists of the root node, branches, internal nodes and the leaf node.
  - The internal nodes are also known as the decision nodes as the conduct evaluations based on the available features.





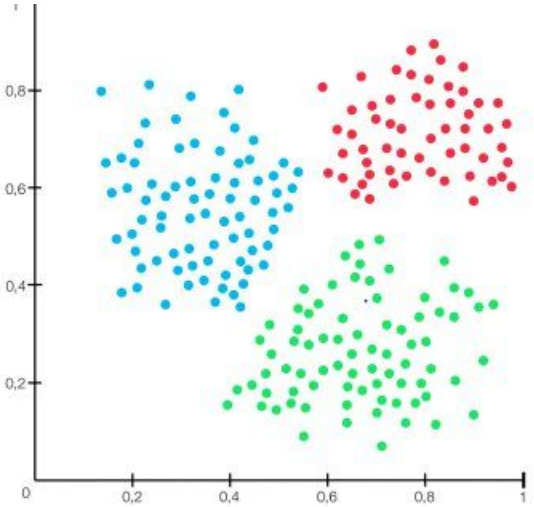
- Linear regression:
  - Linear regression is the technique used to find the linear relationship between two variables. It is the process to determine the best-fit line that describes the relationship between the variables.

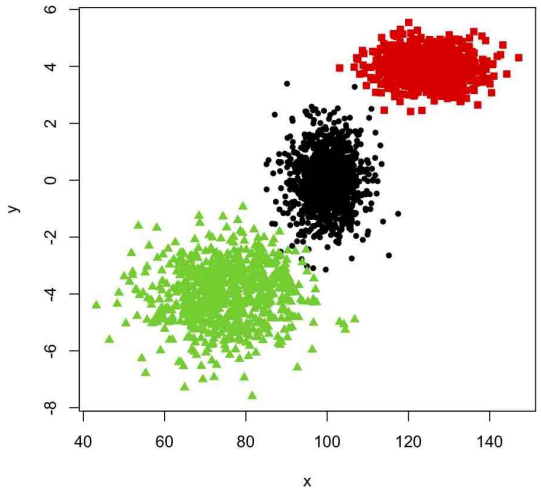


- Logistic regression:
  - It predicts the categorical dependent variable using the given set of input variables.
  - The output is a probabilistic value that lie between 0 to 1.

Unsupervised Learning

- Clustering:
  - Clustering is used for unsupervised learning.
  - It is the process of dividing the data points into number of groups such that data in each groups have similarity between them.

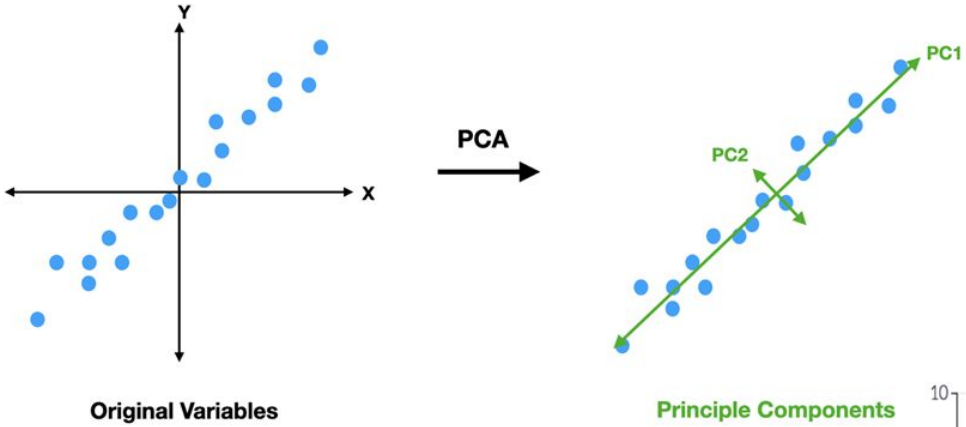


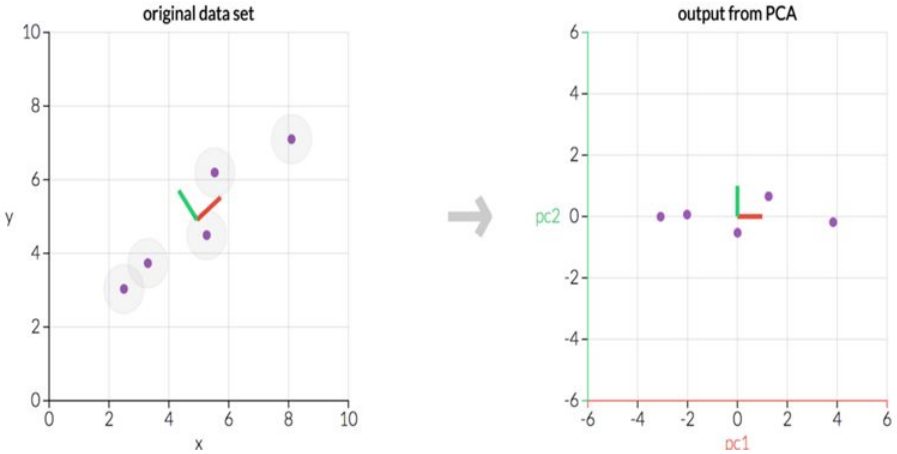


Principal component analysis:

- PCA is an algorithm that is used for dimensionality reduction, by increasing the interpretability and minimising the information loss at the same time.
- It is a process that converts observation of correlated features into a set of linearly uncorrelated features with the help of orthogonal transformation. The new transformed features obtained are called as principal components.
- PCA works by considering the variance of each attribute as high variance shows the good split between the classes.

Examples of PCA

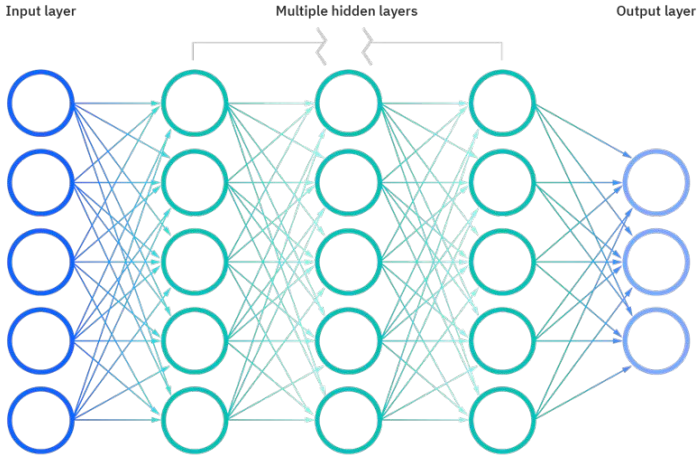




Neural Networks

Neural networks

- Neural networks are a subset of ML and the core of deep learning.
- Human brain cells, called neurons, form a complex, highly interconnected network and send electrical signals to each other to help humans process information.
- Similarly, an artificial neural network is made of artificial neurons that work together to solve a problem.

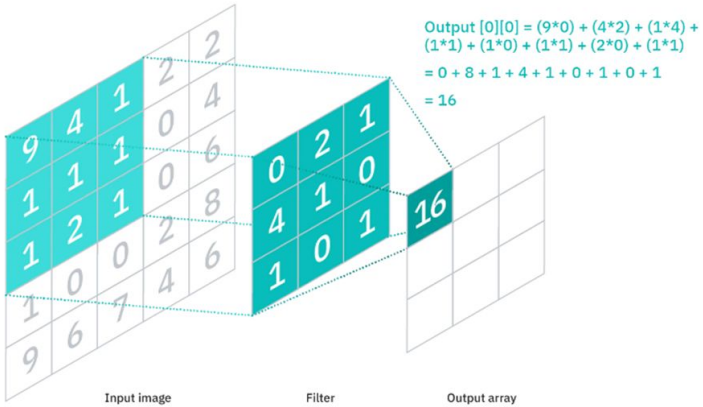


Convolutional neural network

- Convolutional neural network or CNN is a type of neural network used primarily for image recognition and processing of pixel data, due to its ability to recognise patterns in an image.
- The neurons are sparsely connected.
- CNN mainly consists of three types of layers:
  - Convolutional layer
  - Pooling layer
  - Fully connected layer.

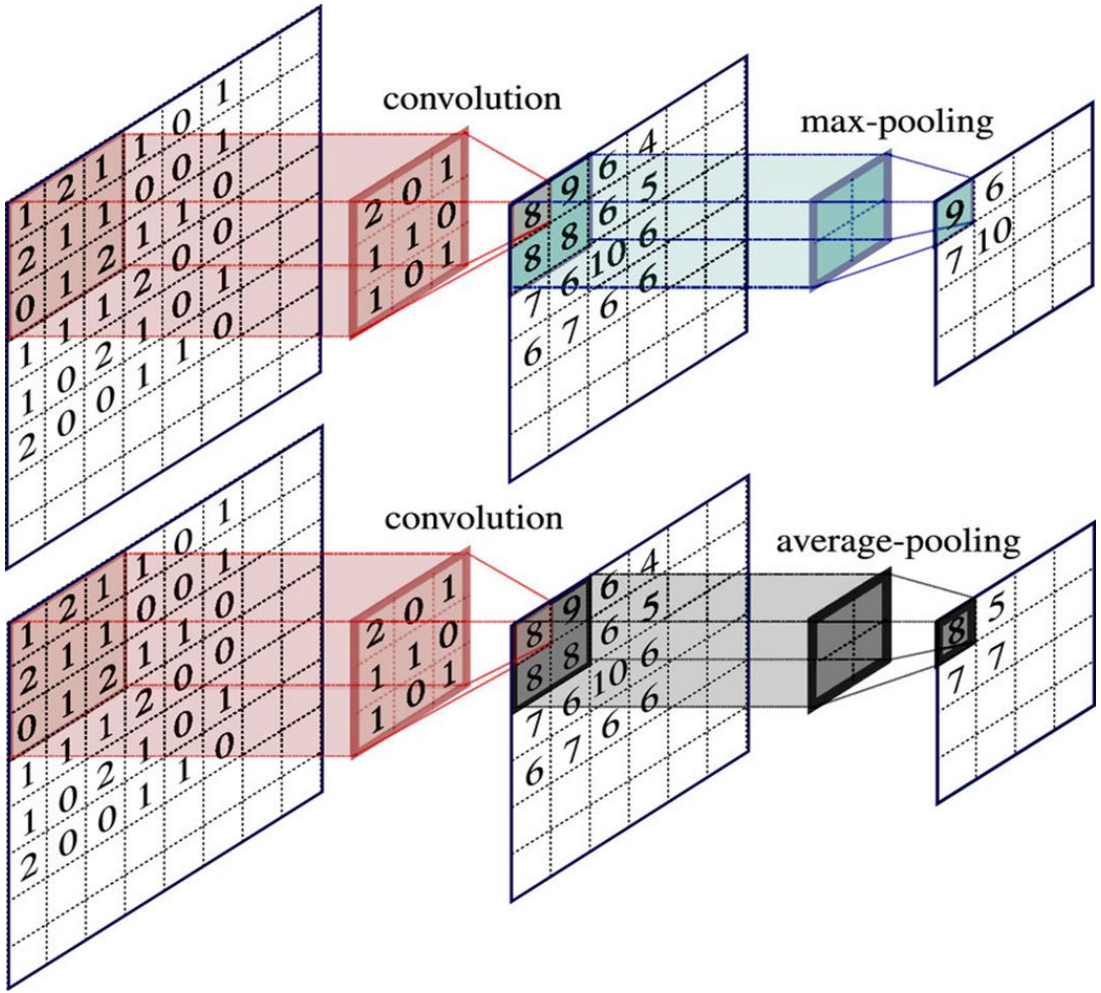
Convolutional layer

- The Convolutional layer requires input data, filter and a feature map.
- The pixel values in the input data is convolved with the values present in the filter.
- The window on the input layer sweeps across the entire input and gives a new value which is mapped on the feature map.
- The feature map is also known as activation map or convolved feature.



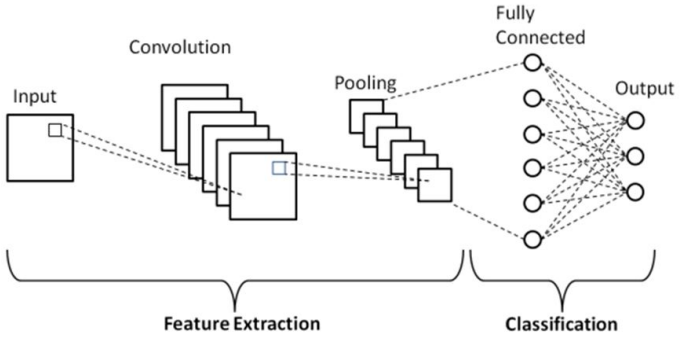
Pooling layer

- Pooling layer down samples and performs dimensionality reduction, reducing the number of parameters in the input.
- There are two types of pooling:
- **Max pooling:** it selects the pixel with the maximum value to send to the output array.
- **Average pooling:** it calculates the average value within the receptive field to send to the output array.


convolution
max-pooling
convolution
average-pooling

Fully connected layer

- In the Fully connected layer or the dense layer, each node in the output layer is connected directly to the node in the previous layer.
- Classification based on the features extracted through the previous layers are done here.
- Softmax activation function is used to classify inputs producing probability from 0 to 1.



Data Preparation

Data Crawling, Data Cleaning, Data Preprocessing

Why do we need Data?

- No ML without data!
- Good data translates to good models.
- Good models translates to better generalization (real world performance).
- Types of Data:
  - Training Data: Data used to train our model
  - Test Data: Data used to evaluate our model
- Why do we need separate training and test data?
  - To prevent problem of overfitting
  - Train the model on training set, evaluate the model on unseen test data
  - Good model results in good performance on test data

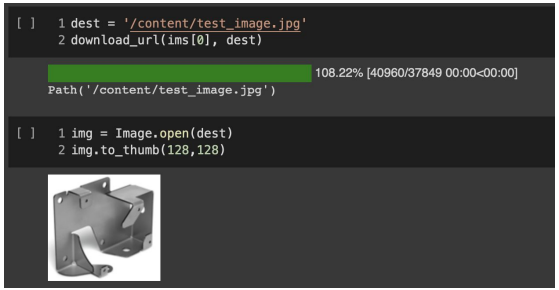
Overfitting and Underfitting

- Overfitting
  - Model is trained too closely to the training data
  - Memorization of training data
  - Model does not generalize well

- Underfitting
  - Model doesn’t even fit the training data properly
  - Training Loss is high
  - Changing the model is necessary

Data Crawling - Sheet Metal Parts

- Bing Image Search API
  - Used `fastai` API to crawl images using the key generated at Microsoft Azure.



- Also used a BBID Script
  - Used an open source BBID (Bulk Bing Image Downloader) Script available at this github link.

Data Crawling - Arbitrary Images

- Unsplash Random Images Collection
  - Kaggle open source Dataset available at this link.

- Manually Crawled Images
  - Manually crawled images from Google Images, Bing Image Search, Unsplash.

Data Cleaning

Data Cleaning was done manually.

- Separating Images with watermarks to the test folder
- Deleting the non-sheet metal garbage images crawled from web
- Preparing the test dataset

Data Preprocessing

- Convert to ‘Tensor’.
- Random Resize images.
- Random Crop images to have size (224,224,3).
- Each pixel stores brightness between 0 and 255.
- Normalize the images.
- Feed it into the model for training.

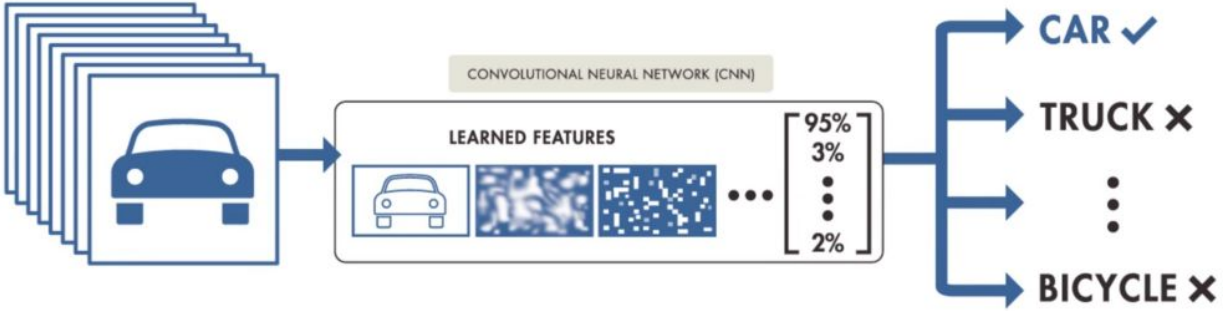
Transfer Learning

What is Transfer Learning?

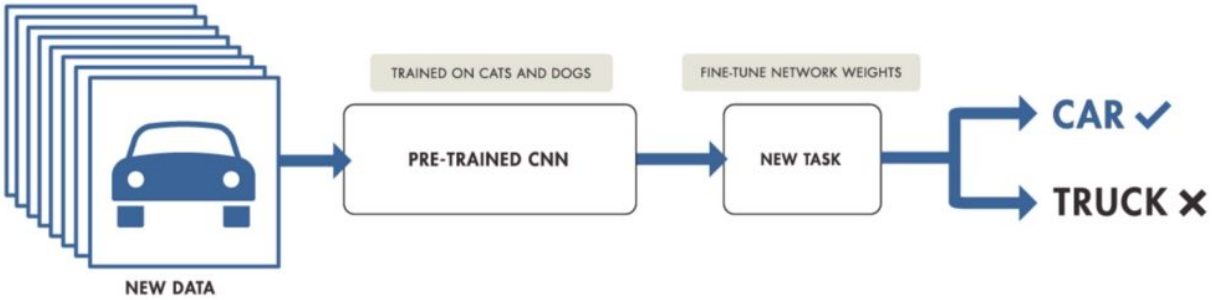
- What is a pre-trained model?
  - A ML model that has already been trained on a large dataset and can can be used for different but related task.
  - Advantages:
    - Saves time and resources
    - Prevents overfitting
    - Doesn’t need too much data

- Uses pre-trained model after fine-tuning on training data

TRAINING FROM SCRATCH

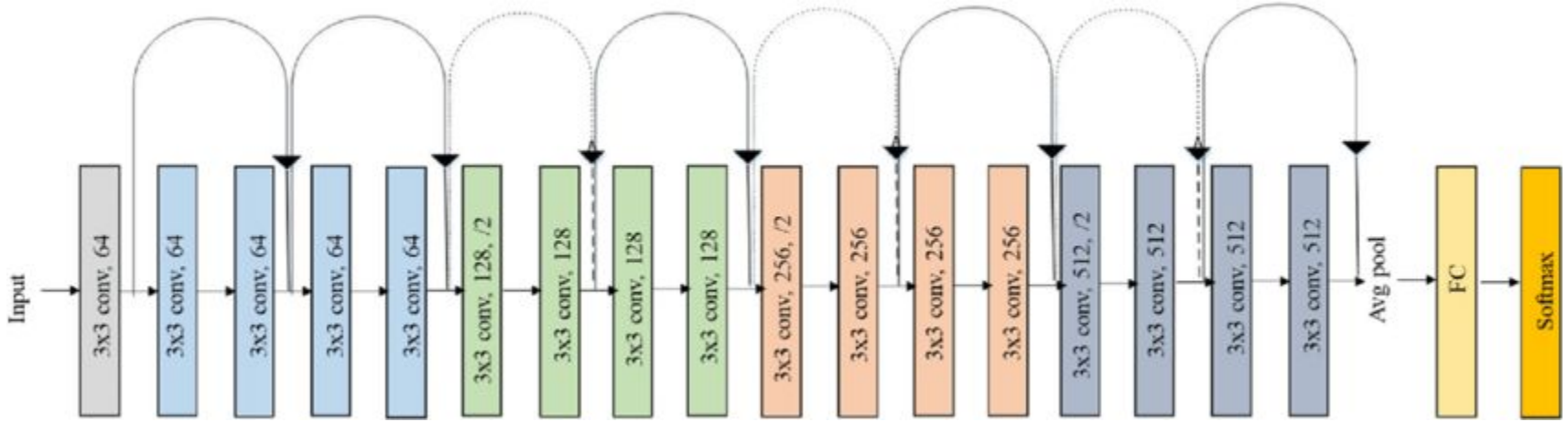


TRANSFER LEARNING

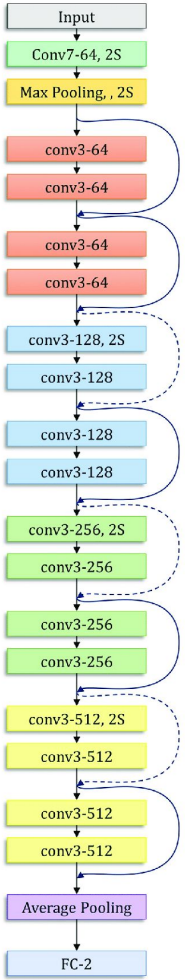


Why transfer learning?

- Works on small datasets.
- Doesn’t need high computational power
- We use ResNet-18 pre-trained model:
  - Trained on ImageNet dataset: over 14 million images organized into 1000 classes
  - Has 18 layers in total, including the Convolution, Pooling and Fully Connected Layers



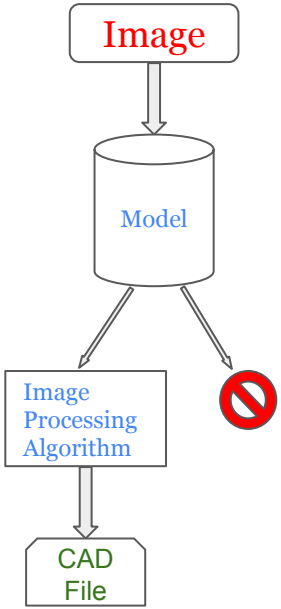
Layer Name	Output Size	ResNet-18
conv1	$112 \times 112 \times 64$	7×7, 64, stride 2
conv2_x	$56 \times 56 \times 64$	3×3 max pool, stride 2 $\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$
conv3_x	$28 \times 28 \times 128$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$
conv4_x	$14 \times 14 \times 256$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$
conv5_x	$7 \times 7 \times 512$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$
average pool	$1 \times 1 \times 512$	7×7 average pool
fully connected	1000	512×1000 fully connections
softmax	1000	



How is useful for us?

The Problem, Our Approach and Hurdles:

- The Second Step was to choose the correct architecture for our classifier model.
- We chose to use Transfer Learning since Pre-Trained architectures are significantly deeper and trained on humongous datasets and are therefore, more robust.
- The obvious candidate was the ResNet models!

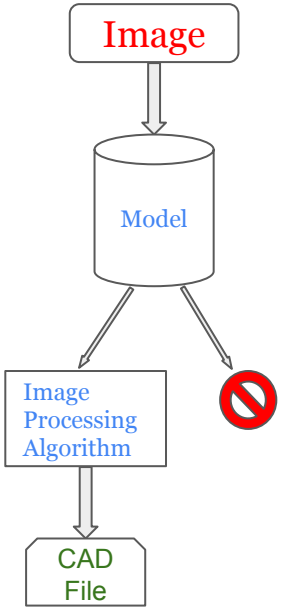


The Problem, Our Approach and Hurdles:

- The outline of the problem at hand is :
  - Given an image, classify it as a sheet metal part or not.
  - If yes, output the corresponding CAD File.

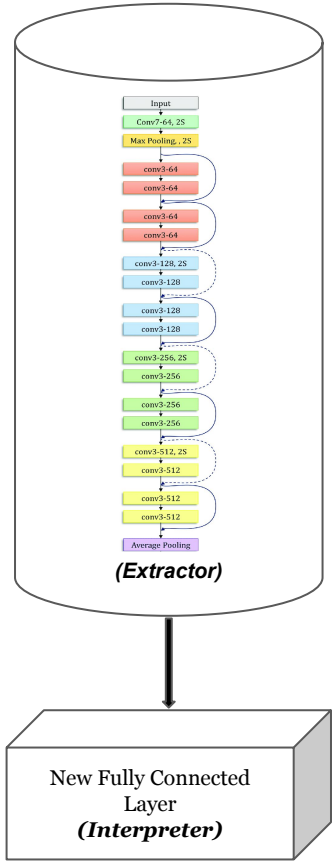
- The pipeline of our algorithm is given here :

- Our First Step was learning Image Processing techniques.



Why ResNet and How ResNet:

- Data is limited; hence, training from scratch is infeasible.
- Transfer Learning lets us harness the power of Very Deep Networks for cheap!
- Fix all but the last layer.
- The last layer can intuitively be thought of as the "*interpreter*" and the previous layers the "*extractors*". Therefore, if the extractor is well trained, it is enough to retrain the interpreter



Other Approaches:

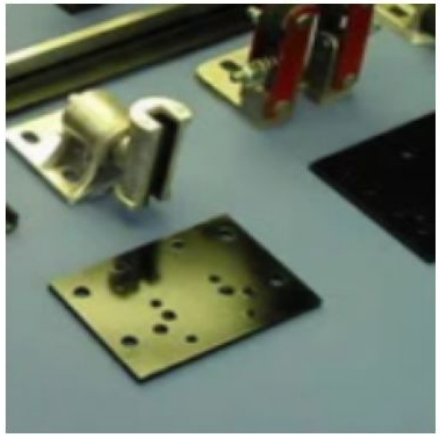
- A slightly different approach we explored was using the ResNet architecture as a feature extractor, but using other classical ML techniques as the interpreter : PCA + Decision Trees/Gaussian Mixture Models

- The performance (accuracy in our case), took a huge hit :
  - Limited Data
  - ResNet is already a very good feature extractor

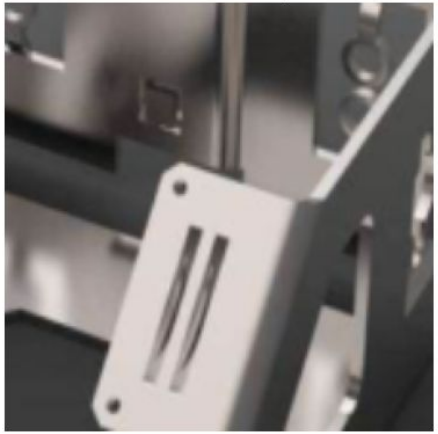
Prediction: non_smp

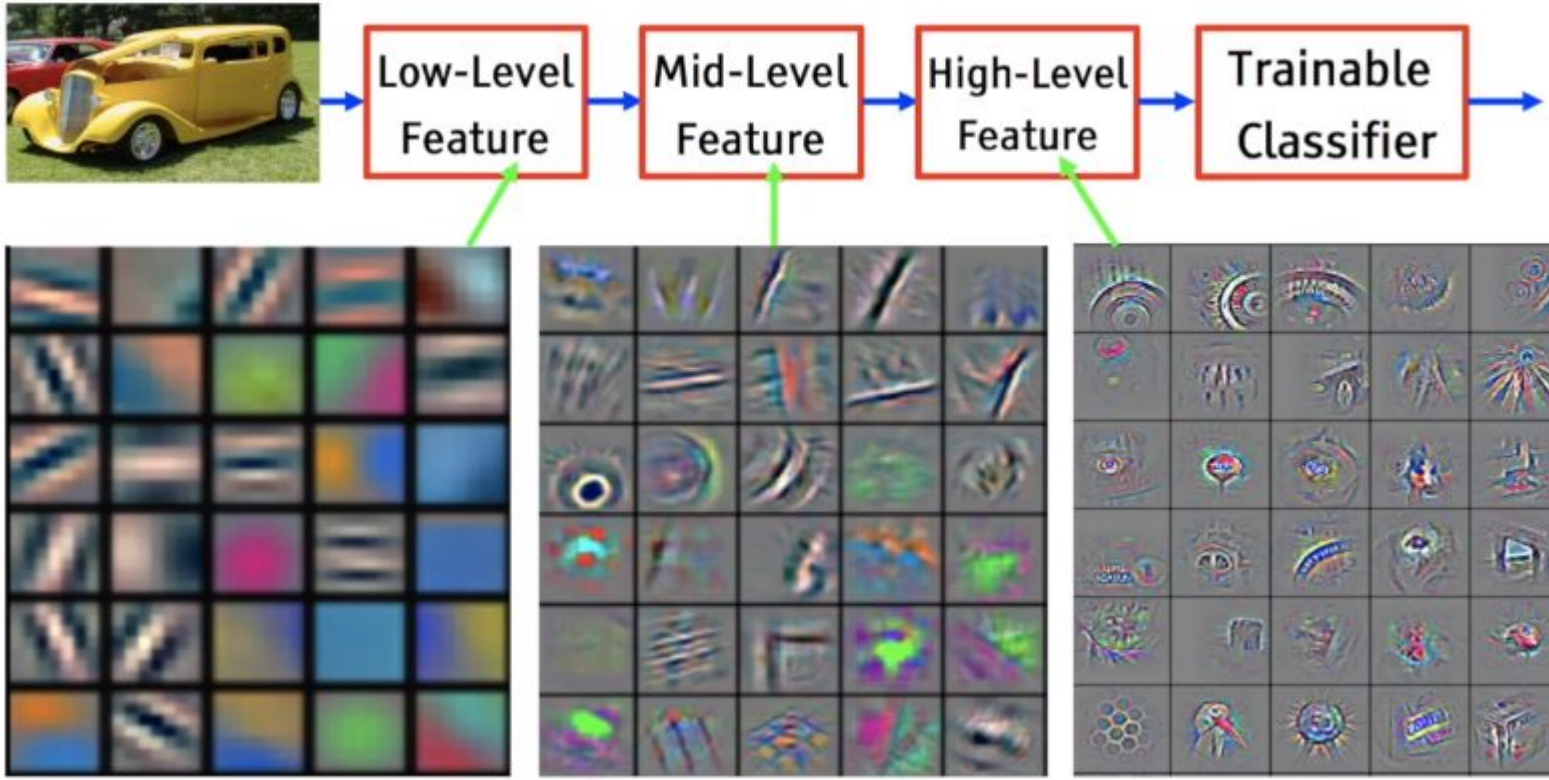
Prediction: non_smp

Prediction: smp



Prediction: smp




Low-Level Feature
Mid-Level Feature
High-Level Feature
Trainable Classifier

THANK YOU!