**Major Project Report**

**On**

**“Signature-Detection”**

**Submitted in partial fulfillment of the requirement for the degree of**

**Degree Name….**

**Session 20…-20…**

**SUBMITTED BY**

**NAME**

**…………**

**Enrollment- ----**

**SUBMITTED TO**

**University Name**

**UNIVERSITY LOGO**

**College name**

**(Approved by …….., Affiliated to ………. University, ……….)**

**CERTIFICATE**

This is to certify that the project titled **“Signature Detection”** isthe bonafide work carried out by **STUDENT NAME,** student of **STREAM** of **UNIVERSITY NAME** in fulfillment of the requirements for the award of the degree of **DEGREE NAME**.

The report submitted by the student is the work of student himself and he/she has fulfilled the requirements of completing his/her work during the training. His/her conduct was good during the entire training period. He/ She were sincere towards his work and did all the study with thorough understanding and dedication.

**(----------------------) (Prof.----------------)**

**Director** **Internal Guide**

**DECLARATION**

I Beg **STUDENT NAME** hereby declare that the Project Report entitled **“PROJECT NAME”** submitted by me to **UNIVESITY NAME** is record of an original work done by me under the guidance of **TRAINER’S NAME**.

This Project report has been submitted in partial fulfillment of the requirement for the award of degree of **DEGREE NAME**. This is my original work and the conclusions drawn therein are based on the material collected by myself. This project report has not been submitted to any other university or institute for the award of any degree or diploma.

**Under the Guidance of: Signature of the Student**

PROF. NAME STUDENT NAME

Uni. Roll no.

Class: - stream & sem

…………………….

(Project Supervisor)

**ACKNOWLEDGEMENT**

*Heartfelt thanks to the following people…*

A Few typewritten words of thanks can-not really express the sincerity of my gratitude. But I am trying to put into words my gratefulness towards all who have helped & encourage me in carrying out the project.

I would like to thank my trainer **TRAINER’S NAME** who helped me throughout the project.

I would then like to thank my faculty guide, Prof. **NAME**, for all his valuable inputs and constant support towards me throughout my project and providing me an opportunity to learn outside the class room. It was a truly wonderful learning experience.

I would like to dedicate this project to my parents. Without their help and constant support this project would not have been possible. I would like to thank all my friends who did their valuable suggestions and support.

Last but not the least I would like to thank all the respondents who offered their opinions and suggestions and sometimes critical views throughout the survey which made me constantly update myself come out with a successful project.

**Roll No: ………………. Student Name: ……………..**

**INDEX**

|  |  |
| --- | --- |
| **S.no** | **Topic** |
| 1. | Introduction |
| 2. | Problem Statement |
| 3. | Objectives |
| 4. | Methodogy |
| 5. | Research Work |
| 6. | METHOLODOGY |
| 7. | Process of Machine Learning |
| 8. | Python Programming |
| 9. | Jupyter Notebook |
| 10. | Machine Learning on Prediction Software |

**Abstract**

In this project, we were asked to experiment with a real world dataset, and to explore how machine learning algorithms can be used to find the houses prices in data. We were expected to gain experience using a common data-mining and machine learning library, and were expected to submit a report about the dataset and the algorithms used. After performing the required tasks on a dataset of my choice, herein lies my final report

**Keywords:**

* **Machine Learning,**
* **Prediction,**
* **Keras and Tensorflow,**
* **Supervised learning**

# Introduction to Python

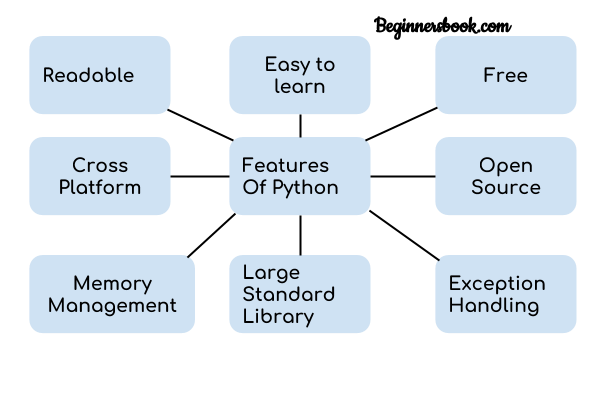
Python is a popular programming language. It was created by Guido van Rossum, and released in 1991.

It is used for:

* Web Development (server-side),
* Software Development,
* Mathematics,
* System Scripting.

**Interesting fact**: Python is named after the comedy television show Monty Python’s Flying Circus. It is not named after the Python snake.

### Why Python?

* Python works on different platforms. (Windows, Mac, Linux, Raspberry Pi etc.)
* Python has a simple syntax similar to the English language.
* Python has syntax that allows developers to write programs with fewer lines than some other programming languages.
* Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.
* Python can be treated in a procedural way, an object-oriented way or a functional way.

**Features**

**Readable:**

 Python is a very readable language.

**Easy to Learn:**

Learning python is easy as this is a expressive and high level programming language, which means it is easy to understand the language and thus easy to learn.

**Cross platform**

Python is available and can run on various operating systems such as Mac, Windows, Linux, Unix etc. This makes it a cross platform and portable language.

**Open Source:**

Python is an open source programming language.

**Large standard library:**

Python comes with a large standard library that has some handy codes and functions which we can use while writing code in Python.

**Free:**

Python is free to download and use. This means you can download it for free and use it in your application. Python is an example of a FLOSS (Free/Libre Open Source Software), which means you can freely distribute copies of this software, read its source code and modify it.

**Supports exception handling:**

If you are new, you may wonder what is an exception? An exception is an event that can occur during program exception and can disrupt the normal flow of program. Python supports exception handling which means we can write less error prone code and can test various scenarios that can cause an exception later on.

**Advanced features:**

Supports generators and list comprehensions. We will cover these features later.

**Automatic memory management:**

Python supports automatic memory management which means the memory is cleared and freed automatically. You do not have to bother clearing the memory.

### Python Syntax compared to other programming languages

* Python was designed for readability, and has some similarities to the English language with influence from mathematics.
* Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.
* Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for this purpose.

## What Can You Do with Python?

You may be wondering what all the applications of Python are. There are so many applications of Python, here are some of the them.

**1. Web development**

Web framework like Django and Flask are based on Python. They help you write server-side code which helps you manage database, write backend programming logic, mapping URLs etc.

**2. Machine learning**

There are many machine learning applications written in Python. Machine learning is a way to write a logic so that a machine can learn and solve a particular problem on its own. For example, products recommendation in websites like Amazon, Flipkart, eBay etc. is a machine learning algorithm that recognizes user’s interest. Face recognition and Voice recognition in your phone is another example of machine learning.

**3. Data Analysis**

Data analysis and data visualization in form of charts can also be developed using Python.

**4. Scripting**

Scripting is writing small programs to automate simple tasks such as sending automated response emails etc. Such type of applications can also be written in Python programming language.

**5. Game development**

You can develop games using Python.

**6. Desktop applications**

You can develop desktop application in Python using library like TKinter or QT.

**Advantages or Benefits of Python**

The Python language has diversified application in the software development companies such as in gaming, web frameworks and applications, language development, prototyping, graphic design applications, etc. This provides the language a higher plethora over other programming languages used in the industry. Some of its advantages are-

* **Extensive Support Libraries**

It provides large standard libraries that include the areas like string operations, Internet, web service tools, operating system interfaces and protocols. Most of the highly used programming tasks are already scripted into it that limits the length of the codes to be written in Python.

* **Integration Feature**

Python integrates the Enterprise Application Integration that makes it easy to develop Web services by invoking COM or COBRA components. It has powerful control capabilities as it calls directly through C, C++ or Java via Jython. Python also processes XML and other mark up languages as it can run on all modern operating systems through same byte code.

* **Improved Programmer’s Productivity**

The language has extensive support libraries and clean object-oriented designs that increase two to tenfold of programmer’s productivity while using the languages like Java, VB, Perl, C, C++ and C#.

* **Productivity**

With its strong process integration features, unit testing framework and enhanced control capabilities contribute towards the increased speed for most applications and productivity of applications. It is a great option for building scalable multi-protocol network applications.

**Limitations or Disadvantages of Python:**

Python has varied advantageous features, and programmers prefer this language to other programming languages because it is easy to learn and code too. However, this language has still not made its place in some computing arenas that includes Enterprise Development Shops. Therefore, this language may not solve some of the enterprise solutions, and limitations include-

* **Difficulty in Using Other Languages**

The Python lovers become so accustomed to its features and its extensive libraries, so they face problem in learning or working on other programming languages. Python experts may see the declaring of cast “values” or variable “types”, syntactic requirements of adding curly braces or semi colons as an onerous task.

* **Weak in Mobile Computing**

Python has made its presence on many desktop and server platforms, but it is seen as a weak language for mobile computing. This is the reason very few mobile applications are built in it like Carbon Nelle.

* **Gets Slow in Speed**

Python executes with the help of an interpreter instead of the compiler, which causes it to slow down because compilation and execution help it to work normally. On the other hand, it can be seen that it is fast for many web applications too.

* **Run-time Errors**

The Python language is dynamically typed so it has many design restrictions that are reported by some [Python developers](http://www.mindfiresolutions.com/python-development.htm). It is even seen that it requires more testing time, and the errors show up when the applications are finally run.

* **Underdeveloped Database Access Layers**

As compared to the popular technologies like JDBC and ODBC, the Python’s database access layer is found to be bit underdeveloped and primitive. However, it cannot be applied in the enterprises that need smooth interaction of complex legacy data.

**INTRODUCTION**

Machine learning is a sub-domain of computer science which evolved from the study of Data prediction in data, and also from the computational learning theory in artificial intelligence.

It is the first-class ticket to most interesting careers in data analytics today. As data sources proliferate along with the computing power to process them, going straight to the data is one of the most straightforward ways to quickly gain insights and make predictions.

Machine Learning can be thought of as the study of a list of sub-problems, decision making, Regression, deep-learning, inductive logic programming, support vector machines, reinforcement learning, similarity and metric learning, genetic algorithms, sparse dictionary learning, etc. Supervised learning, or classification is the machine learning task of inferring a function from a labeled data . In Supervised learning, we have a training set, and a test set. The training and test set consists of a set of examples consisting of input and output vectors, and the goal of the supervised learning algorithm is to infer a function that maps the input vector to the output vector with minimal error. In an optimal scenario, a model trained on a set of examples will classify an unseen example in a correct fashion, which requires the model to generalize from the training set in a reasonable way. In layman’s terms, supervised learning can be termed as the process of concept learning, where a brain is exposed to a set of inputs and result vectors and the brain learns the concept that relates said inputs to outputs. A wide array of supervised machine learning algorithms are available to the machine learning enthusiast.

**Problem Statement**

**Problems and Issues in Supervised learning:**

Before we get started, we must know about how to pick a good machine learning algorithm for the given dataset. To intelligently pick an algorithm to use for a supervised learning task, we must consider the following factors:

1. Heterogeneity of Data:

Many algorithms like neural networks and support vector machines like their feature vectors to be homogeneous numeric and normalized. The algorithms that employ distance metrics are very sensitive to this, and hence if the data is heterogeneous, these methods should be the afterthought. Decision Trees can handle heterogeneous data very easily.

2. Redundancy of Data:

If the data contains redundant information, i.e. contain highly correlated values, then it’s useless to use distance based methods because of numerical instability. In this case, some sort of Regularization can be employed to the data to prevent this situation.

3. Dependent Features:

If there is some dependence between the feature vectors, then algorithms that monitor complex interactions like Neural Networks and Decision Trees fare better than other algorithms.

4. Bias-Variance Tradeoff:

A learning algorithm is biased for a particular input x if, when trained on each of these data sets, it is systematically incorrect when predicting the correct output for x, whereas a learning algorithm has high variance for a particular input x if it predicts different output values when trained on different training sets. The prediction error of a learned classifier can be related to the sum of bias and variance of the learning algorithm, and neither can be high as they will make the prediction error to be high. A key feature of machine learning algorithms is that they are able to tune the balance between bias and variance automatically, or by manual tuning using bias parameters, and using such algorithms will resolve this situation.

5. Curse of Dimensionality:

If the problem has an input space that has a large number of dimensions, and the problem only depends on a subspace of the input space with small dimensions, the machine learning algorithm can be confused by the huge number of dimensions and hence the variance of the algorithm can be

high. In practice, if the data scientist can manually remove irrelevant features from the input data, this is likely to improve the accuracy of the learned function. In addition, there are many algorithms for feature selection that seek to identify the relevant features and discard the irrelevant ones, for instance Principle Component Analysis for unsupervised learning. This reduces the dimensionality.

6. Over fitting:

The programmer should know that there is a possibility that the output values may constitute of an inherent noise which is the result of human or sensor errors.

**OBJECTIVE**

This course aims at providing an introductory and broad overview of the field of ML with the focus on applications on business dealer. Supervised Machine Learning methods are used in the project to predict prices. Simultaneously, while this project can be taken as a separate project, it serves as a preview of topics that are covered in more details in subsequent modules of the specialization Machine Learning and Regression Learning in business dealer. The goal of Guided Tour of Machine Learning in business is to get a sense of what Machine Learning is, what it is for and in how many different business problems it can be applied to. The course is designed for three categories of students: Practitioners working at financial institutions such as banks, asset management firms or hedge funds Individuals interested in applications of ML for personal day trading Current full-time students pursuing a degree in Finance, Statistics, Computer Science, Mathematics, Physics, Engineering or other related disciplines who want to learn about practical

applications of ML in Finance Experience with Python (including numpy, pandas, and IPython/Jupyter notebooks), linear algebra, basic probability theory and basic calculus is necessary to complete assignments in this project.

Methodology

The approach is based on machine learning techniques: Machine trained using different algorithms (with and without *a priori* information), linear regression, support vector machines regression and extreme learning machines. The results obtained using the different techniques are compared in terms of explanatory capacity and predictive potential, both factors facilitating the development of risk prevention measures. Machine are revealed to be the best all-round technique for this type of study, as they combine a powerful interpretative capacity with a predictive capacity that is comparable to that of the best available techniques. Moreover, the Machine force experts to apply a scientific approach to the construction and progressive enrichment of their models and also enable the basis to be laid for an accident prevention policy that is solidly grounded. Furthermore, the procedure enables better variable definition, better structuring of the data capture, coding, and quality control processes.

**Research work on Tensorflow-and-Keras Technics**

TensorFlow is a software library or framework, designed by the Google team to implement machine learning and deep learning concepts in the easiest manner. It combines the computational algebra of optimization techniques for easy calculation of many mathematical expressions.

## Why is TensorFlow So Popular?

TensorFlow is well-documented and includes plenty of machine learning libraries. It offers a few important functionalities and methods for the same.

TensorFlow is also called a “Google” product. It includes a variety of machine learning and deep learning algorithms. TensorFlow can train and run deep neural networks for handwritten digit classification, image recognition, word embedding and creation of various sequence models.

TensorFlow is a free and open-source software library for machine learning and artificial intelligence. It can be used across a range of tasks but has a particular focus on training and inference of deep neural networks.

TensorFlow was developed by the Google Brain team for internal Google use in research and production.[ The initial version was released under the Apache License 2.0 in Google released the updated version of TensorFlow, named TensorFlow 2.0, in September 2019.

TensorFlow can be used in a wide variety of programming languages, most notably Python, as well as Javascript, C++, and Java.[ This flexibility lends itself to a range of applications in many different sectors.

## Tensorflow Used

### Medical

GE Healthcare used TensorFlow to increase the speed and accuracy of MRIs in identifying specific body parts.Google used TensorFlow to create DermAssist, a free mobile application that allows users to take pictures of their skin and identify potential health complications.Sinovation Ventures used TensorFlow to identify and classify eye diseases from optical coherence tomography (OCT) scans

### Social media

Twitter implemented TensorFlow to rank tweets by importance for a given user, and changed their platform to show tweets in order of this ranking. Previously, tweets were simply shown in reverse chronological order.The photo sharing app VSCO used TensorFlow to help suggest custom filters for photos.

### Search Engine

Google officially released RankBrain on October 26, 2015, backed by TensorFlow.

### Education

InSpace, a virtual learning platform, used TensorFlow to filter out toxic chat messages in classrooms.Liulishuo, an online English learning platform, utilized TensorFlow to create an adaptive curriculum for each student.TensorFlow was used to accurately assess a student’s current abilities, and also helped decide the best future content to show based on those capabilities.

### Retail

The e-commerce platform Carousell used TensorFlow to provide personalized recommendations for customers.The cosmetics company ModiFace used TensorFlow to create an augmented reality experience for customers to test various shades of make-up on their face.

**Research work on Keras Technics**

**Keras** is an open-source software library that provides a Python interface for artificial neural networks. Keras acts as an interface for the TensorFlowlibrary.

Up until version 2.3, Keras supported multiple backends, including TensorFlow  Microsoft Cognitive Toolkit, Theano, and PlaidML As of version 2.4, only TensorFlow is supported. Designed to enable fast experimentation with deep neural networks , it focuses on being user-friendly, modular, and extensible. It was developed as part of the research effort of project ONEIROS (Open-ended Neuro-Electronic Intelligent Robot Operating System),and its primary author and maintainer is François Chollet, a Google engineer. Chollet is also the author of the Xception deep neural network model.

## Features

Keras contains numerous implementations of commonly used neural-network building blocks such as layers, objectives, activation functions, optimizers, and a host of tools to make working with image and text data easier to simplify the coding necessary for writing deep neural network code. The code is hosted on GitHub, and community support forums include the GitHub issues page, and a Slack channel.

In addition to standard neural networks, Keras has support for convolutional and recurrent neural networks. It supports other common utility layers like dropout, batch normalization, and pooling.

Keras allows users to productize deep models on smartphones (iOS and Android), on the web, or on the Java Virtual Machine.  It also allows use of distributed training of deep-learning models on clusters of Graphics processing units (GPU)  and tensor processing units (TPU)

**Process of Machine Learning**

The process of machine learning would be broken down in the 7 steps listed below. In order to illustrate the significance and function of each step, we would be using an example of a simple model. This model would be responsible for differentiating between an apple and an orange. Machine learning is capable of much for complex tasks. However, in order to explain the process in simplistic terms, a basic example is taken to explain the relevant concepts.

**Step: 1**

For the purpose of developing our machine learning model, our first step would be to gather relevant data that can be used to differentiate between the 2 fruits. Different parameters can be used to classify a fruit as either an orange or apple. For the sake of simplicity, we would only take 2 features that our model would utilize in order to perform its operation. The first feature would be the color of the fruit itself and the second one being the shape of the fruit.

Using these features, we would hope that our model can accurately differentiate between the 2 fruits.

## Step:2 Preparing that Data

## Once we have gathered the data for the two features, our next step would be to prepare data for further steps. A key focus of this stage is to recognize and minimize any potential biases in our data sets for the 2 features. First, we would randomize the order of our data for the 2 fruits. This is because we do not want the order to have any bearing on the model’s choices. Furthermore, we would examine our data sets for any skewness towards a particular fruit. This again would help in identifying and rectifying a potential bias as it would mean that the model would be adept at identifying one fruit correctly but may struggle with the other fruit.

## Step: 3 Choosing a Model

The selection of the model type is our next course of action once we are done with the data- centric steps. There are various existing models developed by data scientists which can be used for different purposes. These models are designed with different goals in mind. For instance, some models are more suited to dealing with texts while another model may be better equipped to handle images. With regards to our model, a simple linear regression model is suitable for differentiating between fruits. In this case, type of fruit would be our dependent variable while color of the fruit and shape of the fruit would be the 2 predictors or independent variables.

## Step: 4 Training

At the heart of the machine learning process is the training of the model. Bulk of the “learning” is done at this stage. Here we use the part of data set allocated for training to teach our model to differentiate between the 2 fruits. If we view our model in mathematical terms, the inputs i.e. our 2 features would have coefficients. These coefficients are called the weights of features. There would also be a constant or y-intercept involved. This is referred to as the bias of the model. The process of determining their values is of trial and error. Initially, we pick random values for them and provide inputs. The achieved output is compared with actual output and the difference is minimized by trying different values of weights and biases. The iterations are repeated using different entries from our training data set until the model reaches the desired level of accuracy.

## Step: 5 Evaluation

With the model trained, it needs to be tested to see if it would operate well in real world situations. That is why the part of the data set created for evaluation is used to check the model’s proficiency. This puts the model in a scenario where it encounters situations that were not a part of its training. In our case, it could mean trying to identify a type of an apple or an orange that is completely new to the model. However, through its training, the model should be capable enough to extrapolate the information and deem whether the fruit is an apple or an orange.

**Step : 6 Hyper parameter Tuning**

If the evaluation is successful, we proceed to the step of hyper parameter tuning. This step tries to improve upon the positive results achieved during the evaluation step. For our example, we would see if we can make our model even better at recognizing apples and oranges. There are different ways we can go about improving the model. One of them is revisiting the training step and use multiple sweeps of the training data set for training the model. This could lead to greater accuracy as the longer duration of training provides more exposure and improves quality of the model.

Another way to go about it is refining the initial values given to the model.  Random initial values often produce poor results as they are gradually refined by trial and error. However, if we can come up with better initial values or perhaps initiate the model using a distribution instead of a value then our results could get better. There are other parameters that we could play around with in order to refine the model but the process is more intuitive than logical so there is no definite approach for it.

**Step: 7 Predictions**

The final step of the machine learning process is prediction. This is the stage where we consider the model to be ready for practical applications. Our fruit model should now be able to answer the question whether the given fruit is an apple or an orange. The model gains independence from human interference and draws its own conclusion on the basis of its data sets and training.

The challenge for the model remains whether it can outperform or at least match human judgment in different relevant scenarios.

**Scope of Machine Learning**

The scope of Machine Learning is not limited to the investment sector. Rather, it is expanding across all fields such as banking and finance, information technology, media & entertainment, gaming, and the automotive industry. As the Machine Learning scope is very high, there are some of the areas where researchers are working toward revolutionizing the world for the future.

### Automotive Industry

### Robotics

### Quantum Computing

### Computer Vision

### Health Care

### Python Programming Language

# What is Python

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

Often, programmers fall in love with Python because of the increased productivity it provides. Since there is no compilation step, the edit-test-debug cycle is incredibly fast. Debugging Python programs is easy: a bug or bad input will never cause a segmentation fault. Instead, when the interpreter discovers an error, it raises an exception. When the program doesn't catch the exception, the interpreter prints a stack trace. A source level debugger allows inspection of local and global variables, evaluation of arbitrary expressions, setting breakpoints, stepping through the code a line at a time, and so on. The debugger is written in Python itself, testifying to Python's introspective power. On the other hand, often the quickest way to debug a program is to add a few print statements to the source: the fast edit-test-debug cycle makes this simple approach very effective.

# Applications for Python

Python is used in many application domains. Here's a sampling.

* The Python Package Index lists thousands of third party modules for Python.

# Web and Internet Development

Python offers many choices for web development:

* Frameworks such as Django and Pyramid.
* Micro-frameworks such as Flask and Bottle.
* Advanced content management systems such as [Plone](http://www.plone.org/) and django CMS.

Python's standard library supports many Internet protocols:

* HTML and XML
* JSON
* E-mail processing.
* Support for FTP, IMAP, and other Internet protocol.
* Easy-to-use socket interface.

And the Package Index has yet more libraries:

* Requests, a powerful HTTP client library.
* Beautiful Soup, an HTML parser that can handle all sorts of oddball HTML.
* [Feedparser](https://pypi.org/project/feedparser/) for parsing RSS/Atom feeds.
* [Paramiko](https://pypi.org/project/paramiko/), implementing the SSH2 protocol.
* Twisted Python, a framework for asynchronous network programming.

# Scientific and Numeric

Python is widely used in scientific and numeric computing:

* [SciPy](http://scipy.org/) is a collection of packages for mathematics, science, and engineering.
* [Pandas](http://pandas.pydata.org/) is a data analysis and modeling library.
* [IPython](http://ipython.org/) is a powerful interactive shell that features easy editing and recording of a work session, and supports visualizations and parallel computing.
* The Software Carpentry Course teaches basic skills for scientific computing, running bootcamps and providing open-access teaching materials.

# Education

Python is a superb language for teaching programming, both at the introductory level and in more advanced courses.

* Books such as How to Think Like a Computer Scientist, Python Programming: An Introduction to Computer Science, and Practica Programming.
* The Education Special Interest Group is a good place to discuss teaching issues.

# Desktop GUIs

The [Tk](http://wiki.python.org/moin/TkInter) GUI library is included with most binary distributions of Python.

Some toolkits that are usable on several platforms are available separately:

* [wxWidgets](http://www.wxpython.org/)
* [Kivy](http://kivy.org/), for writing multitouch applications.
* Qt via [pyqt](http://www.riverbankcomputing.co.uk/software/pyqt/intro) or [pyside](http://www.pyside.org/)

Platform-specific toolkits are also available:

* GTK+
* Microsoft Foundation Classes through the [win32 extensions](http://sourceforge.net/projects/pywin32/)

# Software Development

Python is often used as a support language for software developers, for build control and management, testing, and in many other ways.

* [SCons](http://www.scons.org/) for build control.
* [Buildbot](http://buildbot.sourceforge.net/) and Apache Gump for automated continuous compilation and testing.
* Roundup or [Trac](http://www.edgewall.com/trac/) for bug tracking and project management.

# Business Applications

Python is also used to build ERP and e-commerce systems:

* [Odoo](https://www.odoo.com/) is an all-in-one management software that offers a range of business applications that form a complete suite of enterprise management applications.
* [Tryton](http://www.tryton.org/) is a three-tier high-level general purpose application platform.

**Jupyter Notebook**

## Introduction

The notebook extends the console-based approach to interactive computing in a qualitatively new direction, providing a web-based application suitable for capturing the whole computation process: developing, documenting, and executing code, as well as communicating the results. The Jupyter notebook combines two components:

**A web application**: a browser-based tool for interactive authoring of documents which combine explanatory text, mathematics, computations and their rich media output.

**Notebook documents**: a representation of all content visible in the web application, including inputs and outputs of the computations, explanatory text, mathematics, images, and rich media representations of objects.

### Main features of the web application

* In-browser editing for code, with automatic syntax highlighting, indentation, and tab completion/introspection.
* The ability to execute code from the browser, with the results of computations attached to the code which generated them.
* Displaying the result of computation using rich media representations, such as HTML, LaTeX, PNG, SVG, etc. For example, publication-quality figures rendered by the matplotlib library, can be included inline.
* In-browser editing for rich text using the Markdown markup language, which can provide commentary for the code, is not limited to plain text.
* The ability to easily include mathematical notation within markdown cells using LaTeX, and rendered natively by MathJax.

### Notebook documents

Notebook documents contains the inputs and outputs of a interactive session as well as additional text that accompanies the code but is not meant for execution. In this way, notebook files can serve as a complete computational record of a session, interleaving executable code with explanatory text, mathematics, and rich representations of resulting objects. These documents are internally JSON files and are saved with the .ipynb extension. Since JSON is a plain text format, they can be version-controlled and shared with colleagues.

Notebooks may be exported to a range of static formats, including HTML (for example, for blog posts), reStructuredText, LaTeX, PDF, and slide shows, via the nbconvert command.

Furthermore, any .ipynb notebook document available from a public URL can be shared via the Jupyter Notebook Viewer (nbviewer). This service loads the notebook document from the URL and renders it as a static web page. The results may thus be shared with a colleague, or as a public blog post, without other users needing to install the Jupyter notebook themselves. In effect, nbviewer is simply nbconvert as a web service, so you can do your own static conversions with nbconvert, without relying on nbviewer.

### Notebooks and privacy

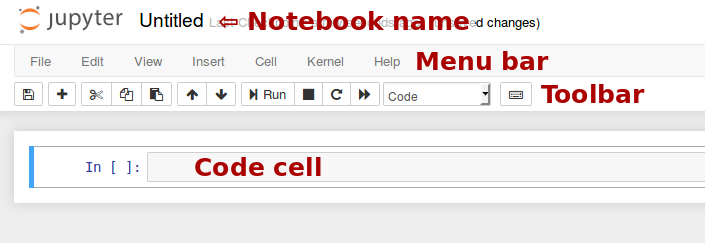
Because you use Jupyter in a web browser, some people are understandably concerned about using it with sensitive data. However, if you followed the standard install instructions, Jupyter is actually running on your own computer. If the URL in the address bar starts with http://localhost: or http://127.0.0.1:, it’s your computer acting as the server. Jupyter doesn’t send your data anywhere else—and as it’s open source, other people can check that we’re being honest about this.

You can also use Jupyter remotely: your company or university might run the server for you, for instance. If you want to work with sensitive data in those cases, talk to your IT or data protection staff about it.

We aim to ensure that other pages in your browser or other users on the same computer can’t access your notebook server. See Security in the Jupyte notebook server for more about this.

## Notebook user interface

When you create a new notebook document, you will be presented with the **notebook name**, a **menu bar**, a **toolbar** and an empty **code cell**.



**Notebook name**: The name displayed at the top of the page, next to the Jupyter logo, reflects the name of the .ipynb file. Clicking on the notebook name brings up a dialog which allows you to rename it. Thus, renaming a notebook from “Untitled0” to “My first notebook” in the browser, renames the Untitled0.ipynb file to My first notebook.ipynb.

**Menu bar**: The menu bar presents different options that may be used to manipulate the way the notebook functions.

**Toolbar**: The tool bar gives a quick way of performing the most-used operations within the notebook, by clicking on an icon.

**Code cell**: the default type of cell; read on for an explanation of cells.

## Structure of a notebook document

The notebook consists of a sequence of cells. A cell is a multiline text input field, and its contents can be executed by using Shift-Enter, or by clicking either the “Play” button the toolbar, or Cell, Run in the menu bar. The execution behavior of a cell is determined by the cell’s type. There are three types of cells: **code cells**, **markdown cells**, and **raw cells**. Every cell starts off being a **code cell**, but its type can be changed by using a drop-down on the toolbar (which will be “Code”, initially), or via [keyboard shortcuts](https://jupyter-notebook.readthedocs.io/en/stable/notebook.html#keyboard-shortcuts).

For more information on the different things you can do in a notebook, see the [collection of examples](https://nbviewer.jupyter.org/github/jupyter/notebook/tree/master/docs/source/examples/Notebook/).

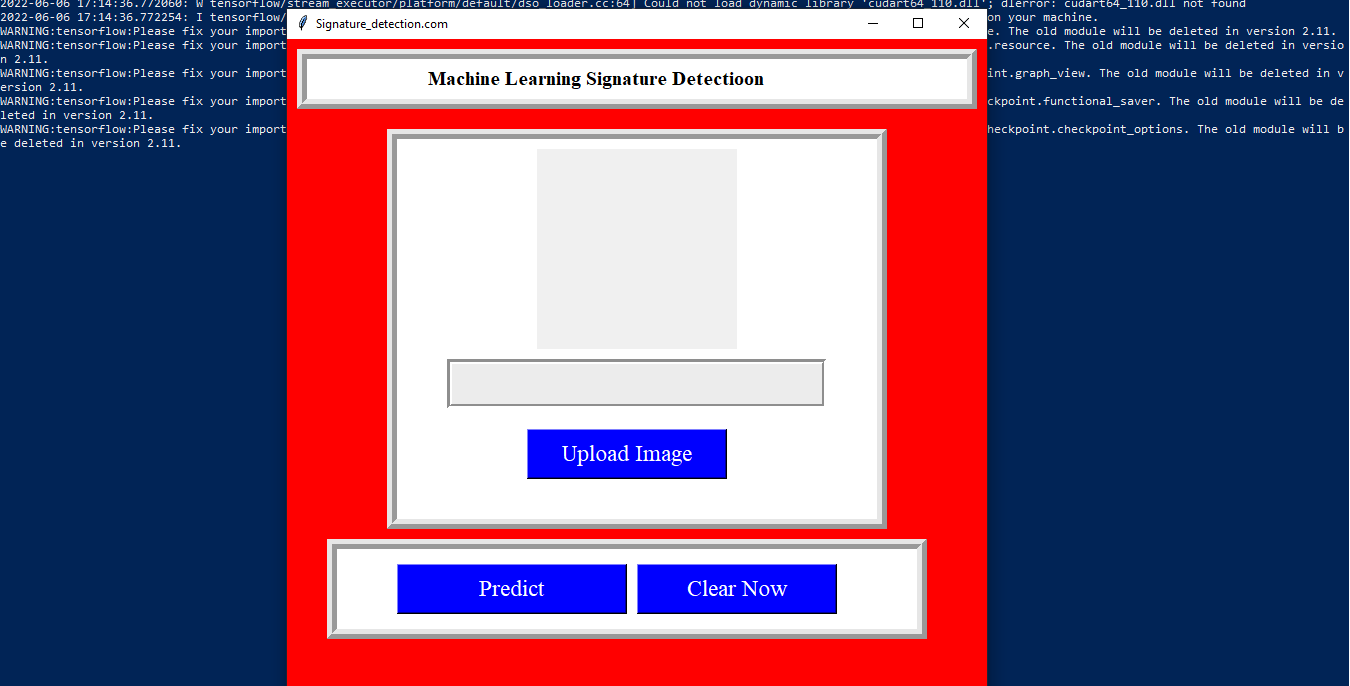
### Code cells

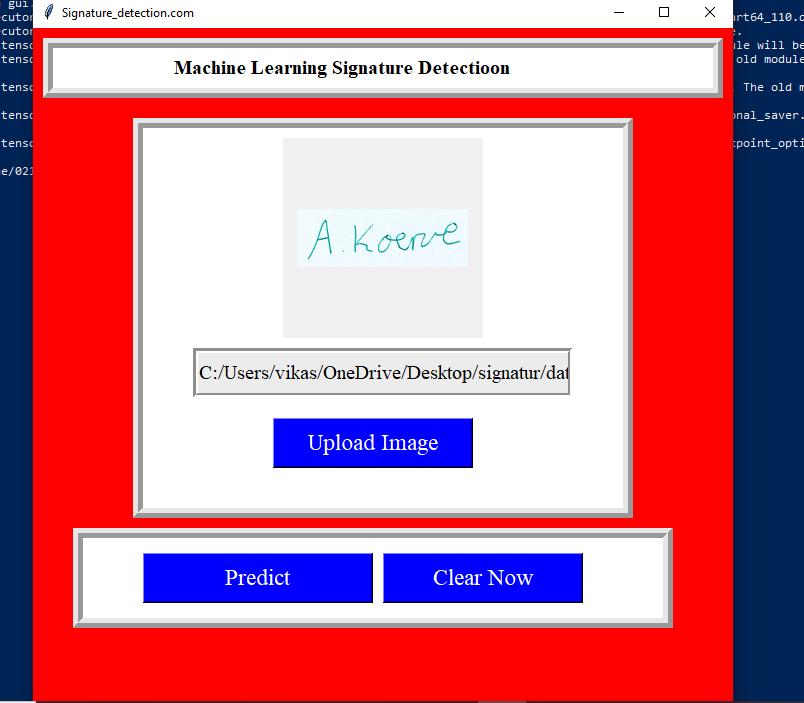
A code cell allows you to edit and write new code, with full syntax highlighting and tab completion. The programming language you use depends on the kernel, and the default kernel (IPython) runs Python code.

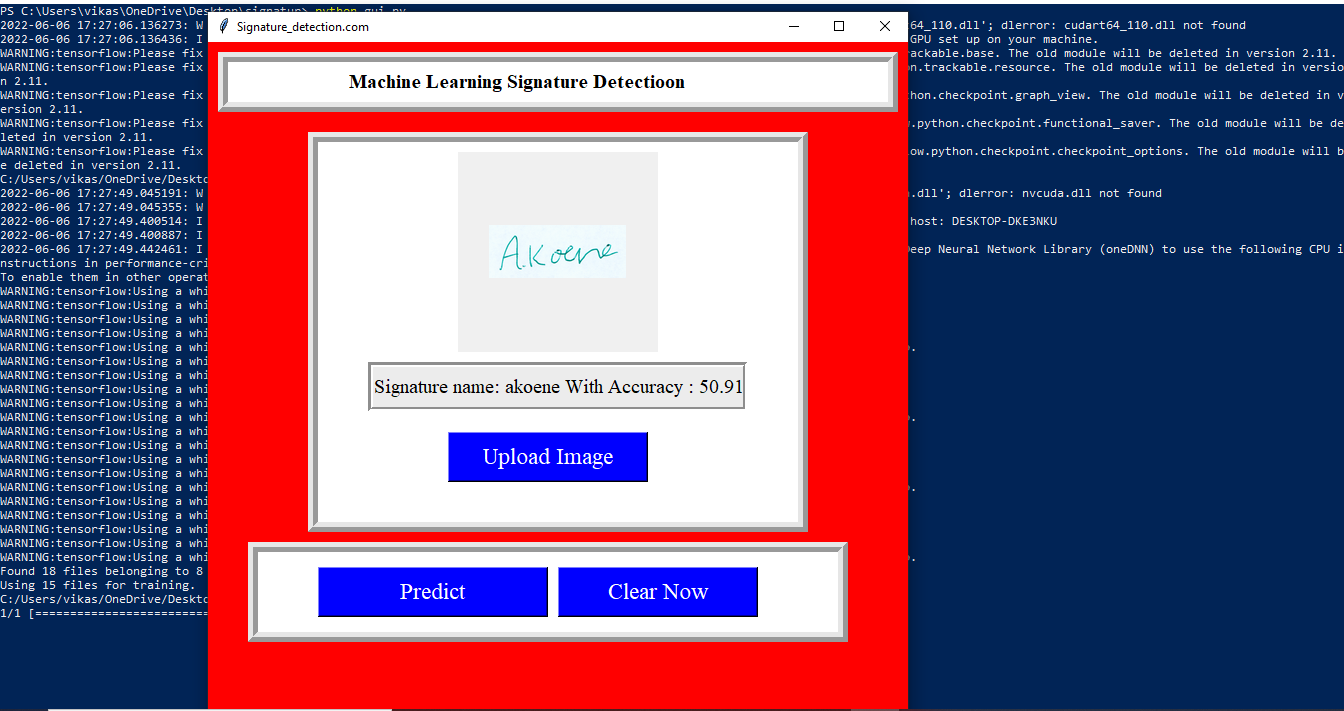
When a code cell is executed, code that it contains is sent to the kernel associated with the notebook. The results that are returned from this computation are then displayed in the notebook as the cell’s output. The output is not limited to text, with many other possible forms of output are also possible, including matplotlib figures and HTML tables (as used, for example, in the pandas data analysis package). This is known as IPython’s rich display capability.

**1.9 Implementation of Signature Detection**

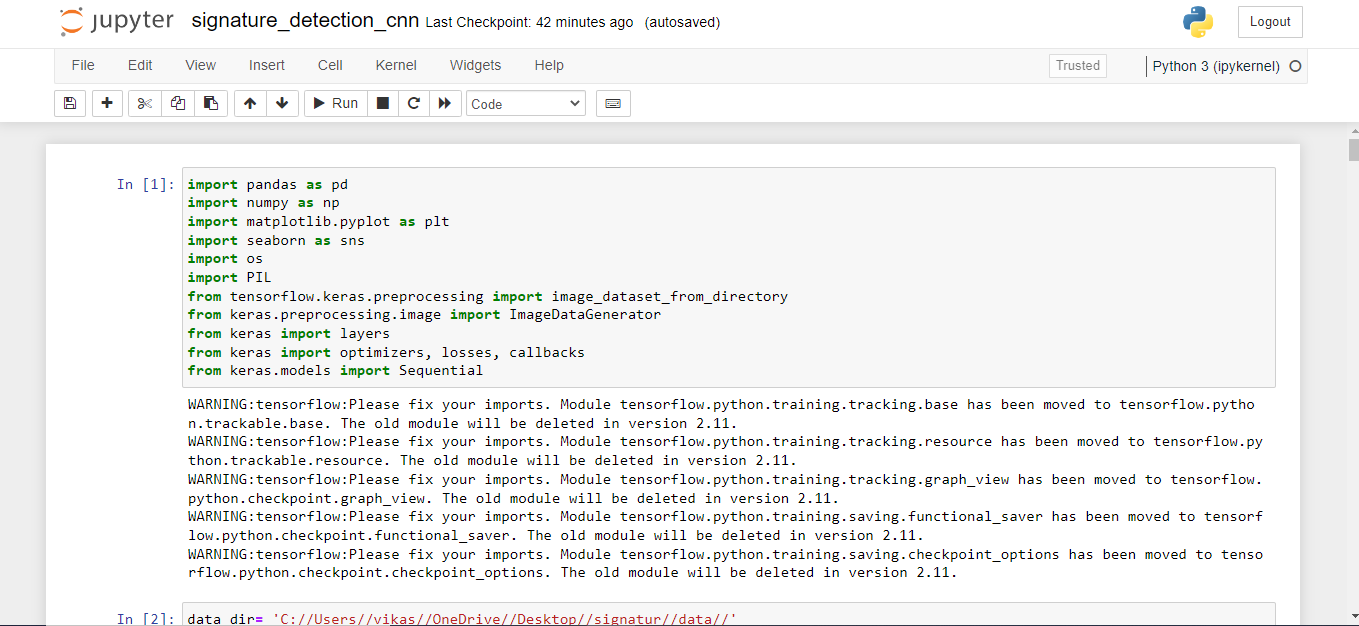
**Foentend :**

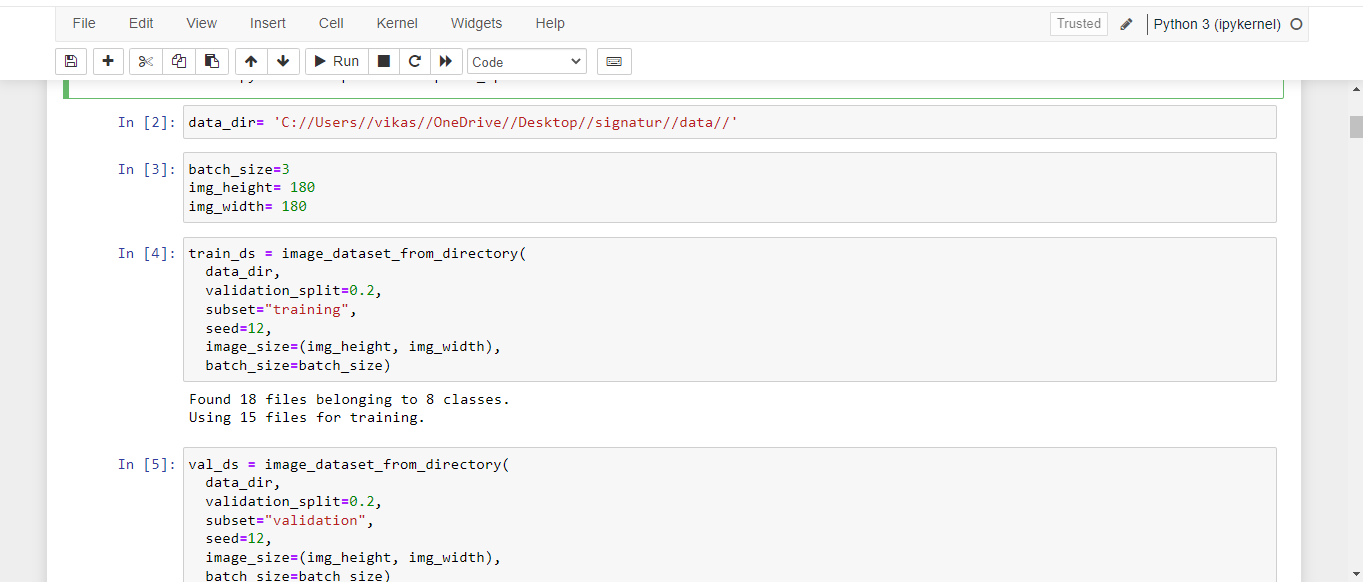
****

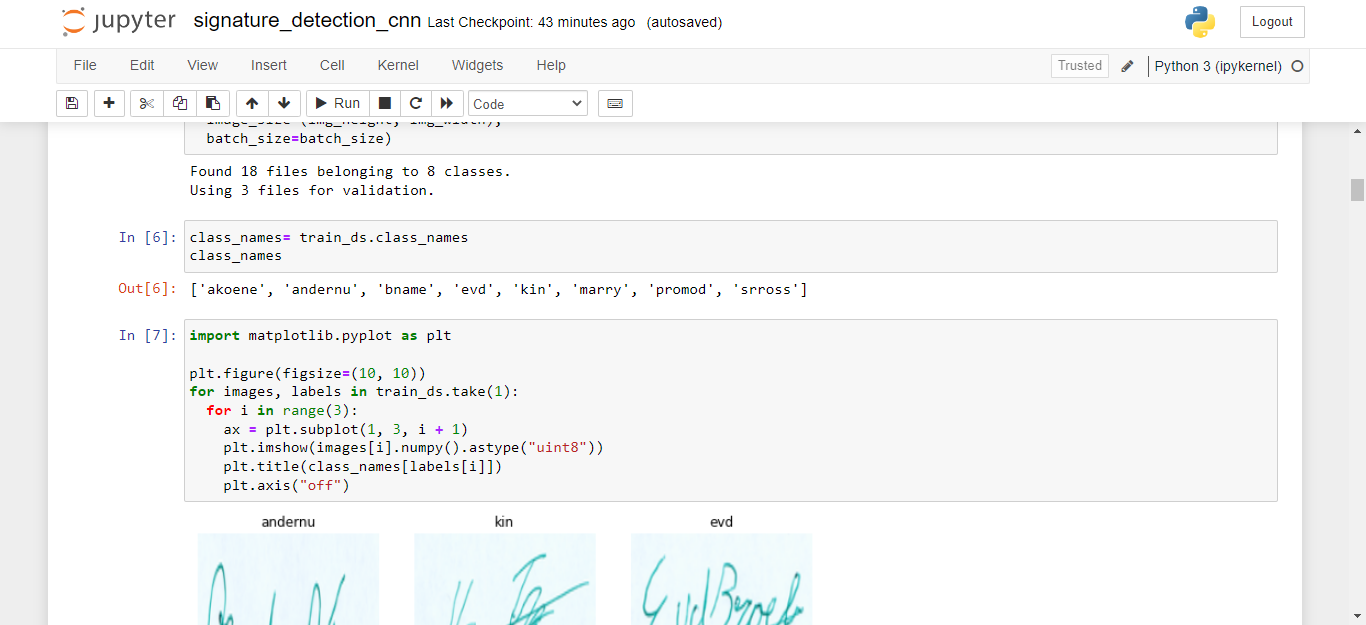
****

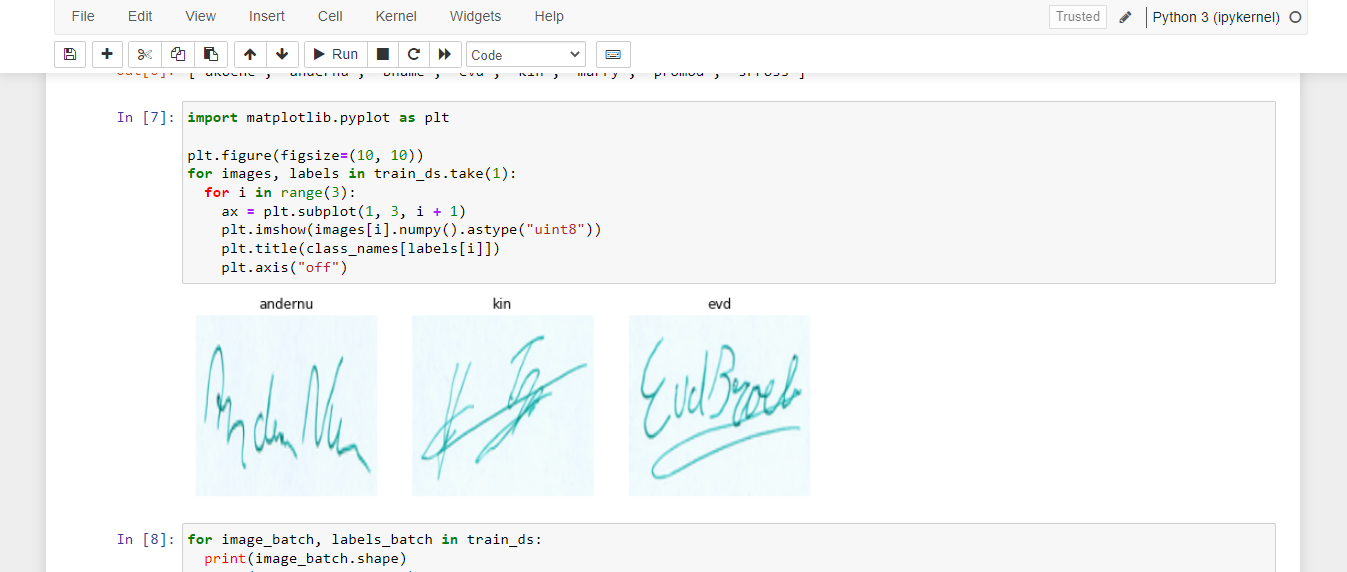
****

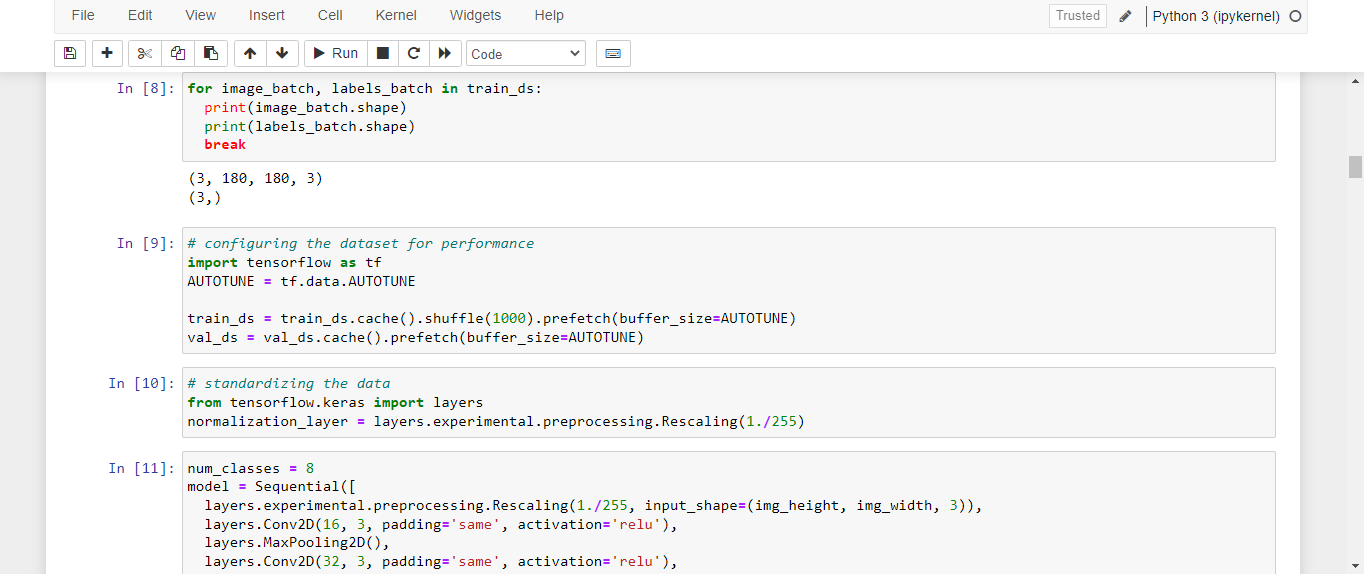
**Backend :**

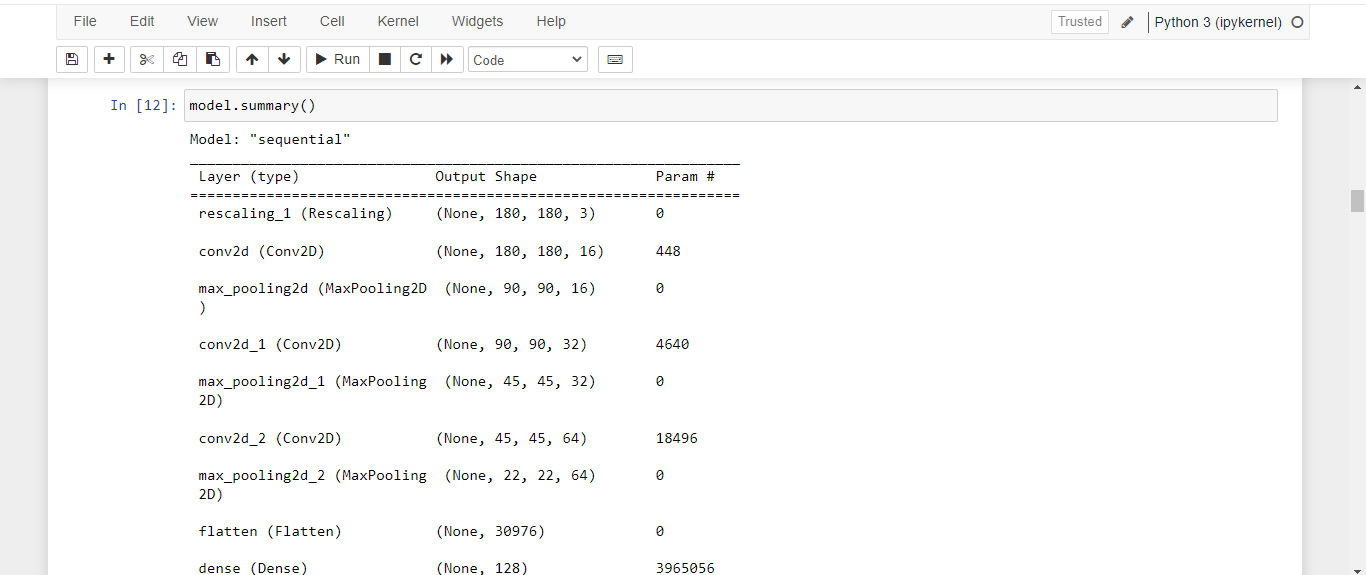


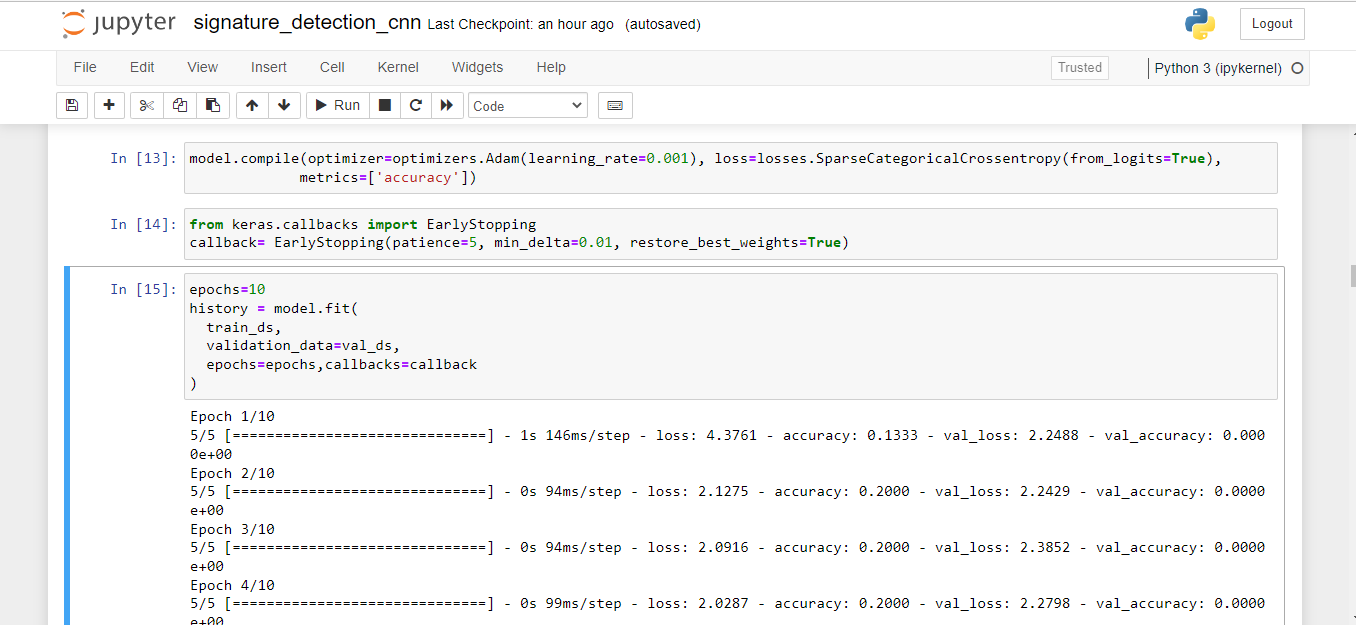


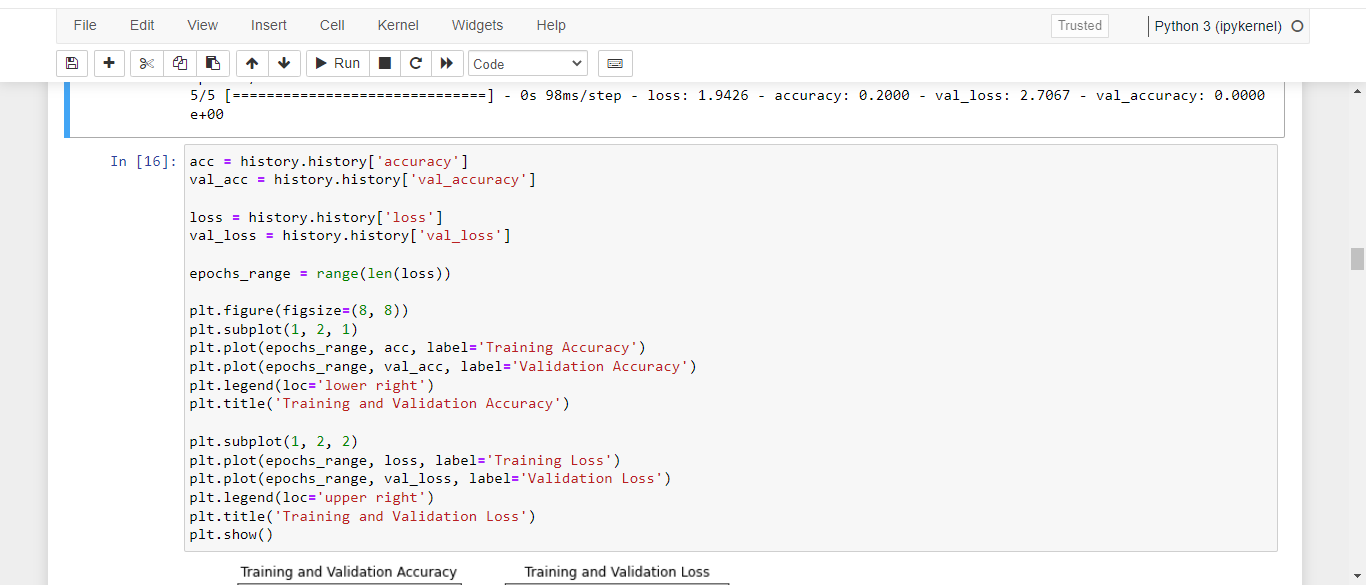


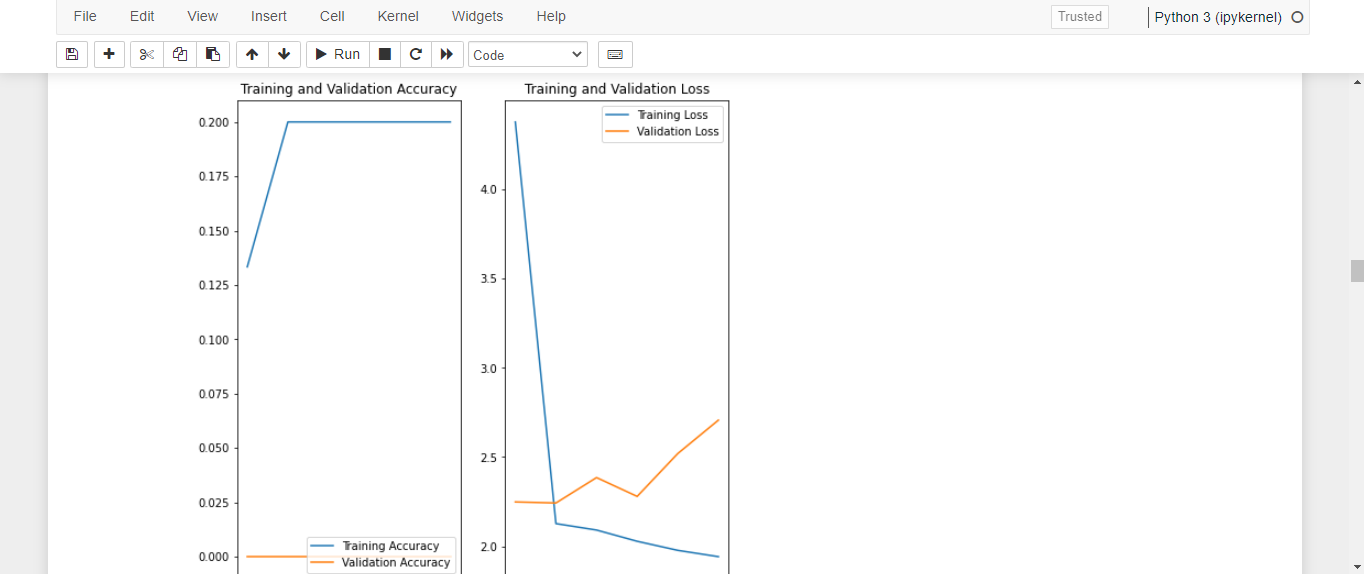


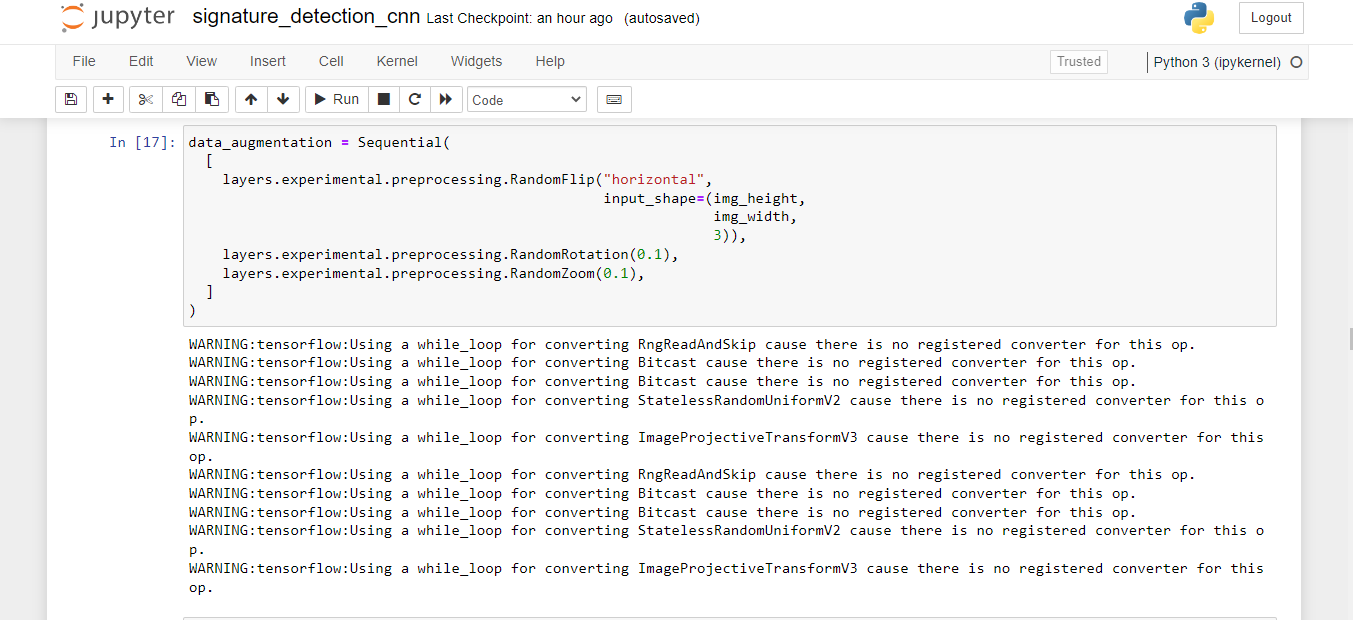


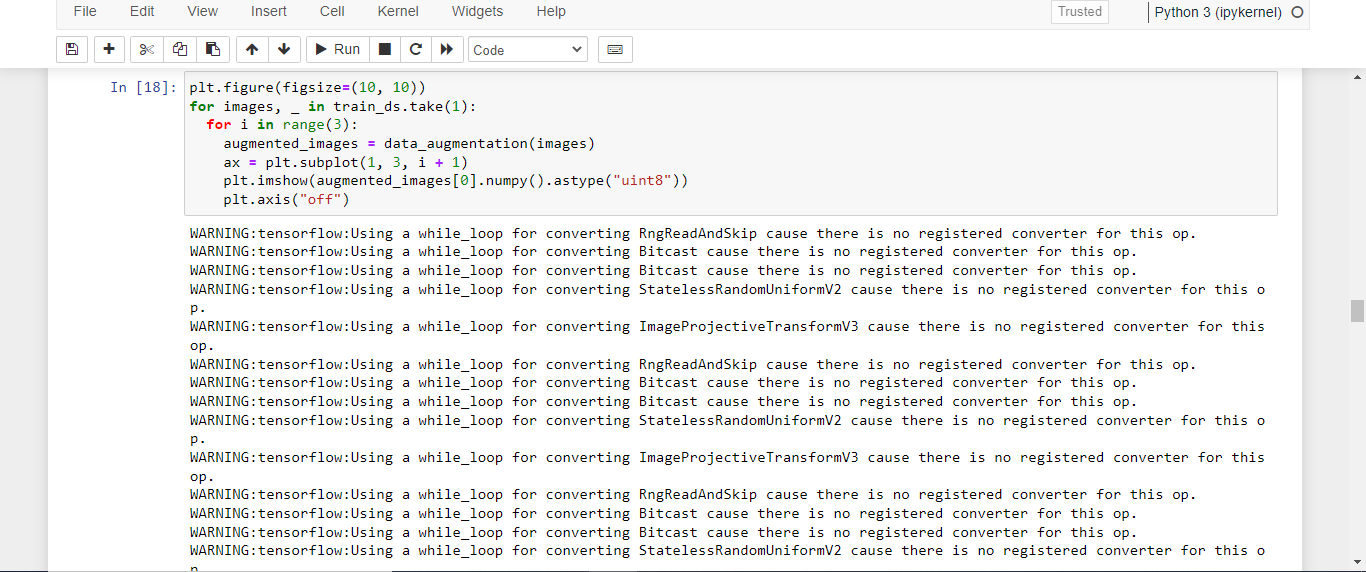


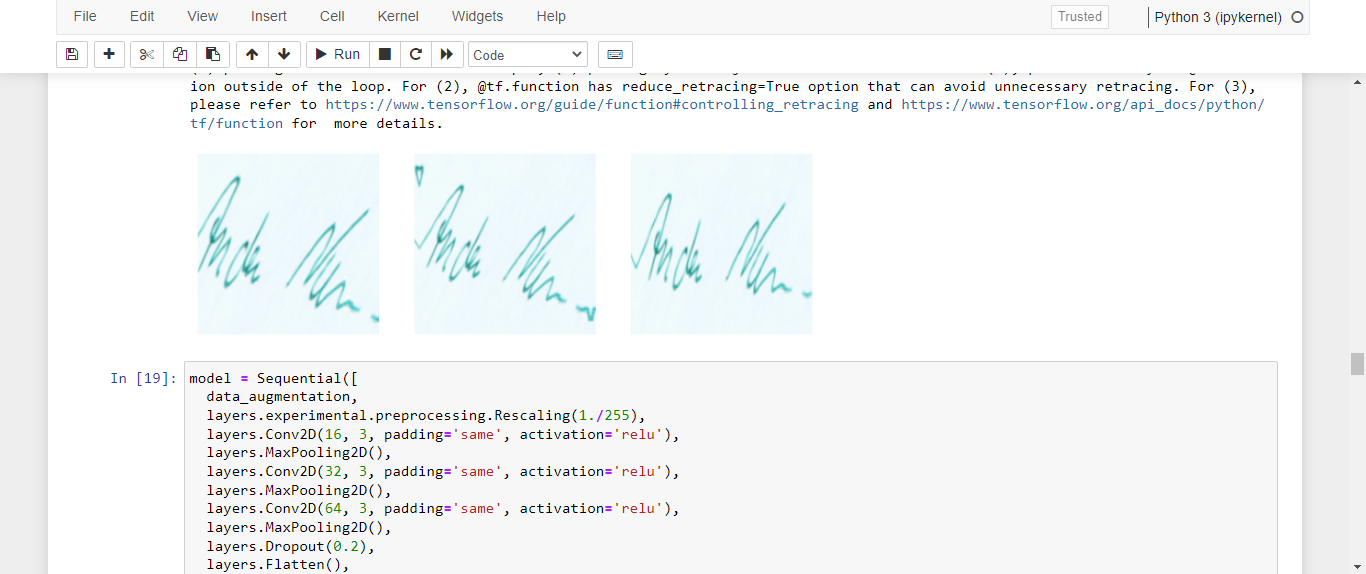


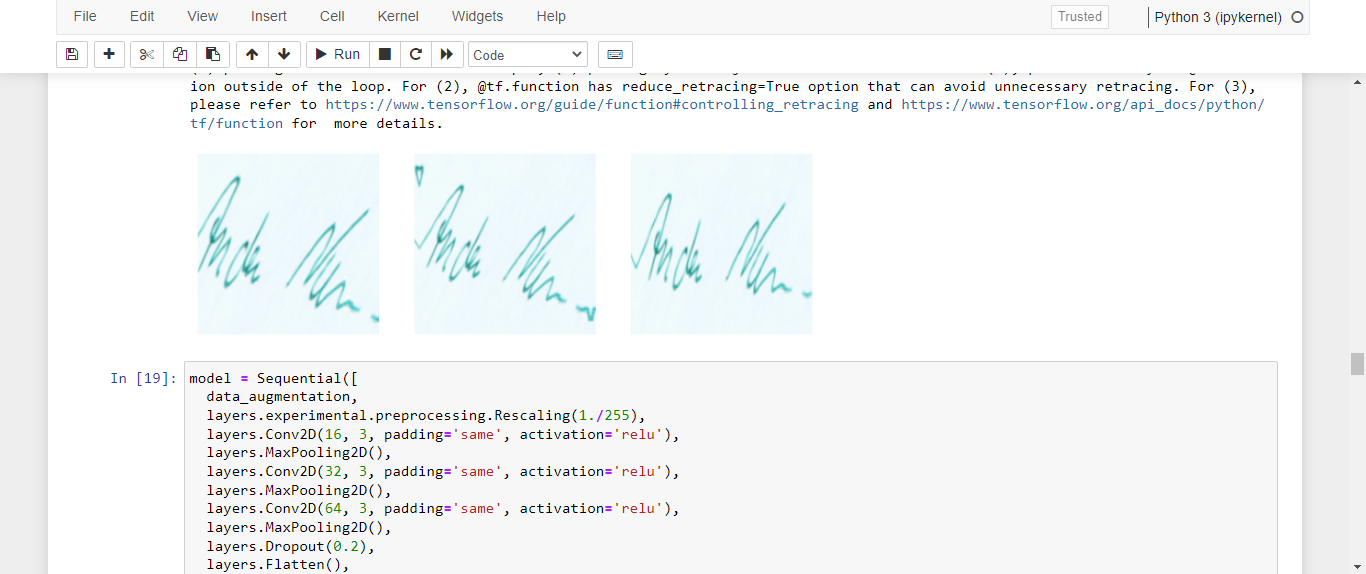


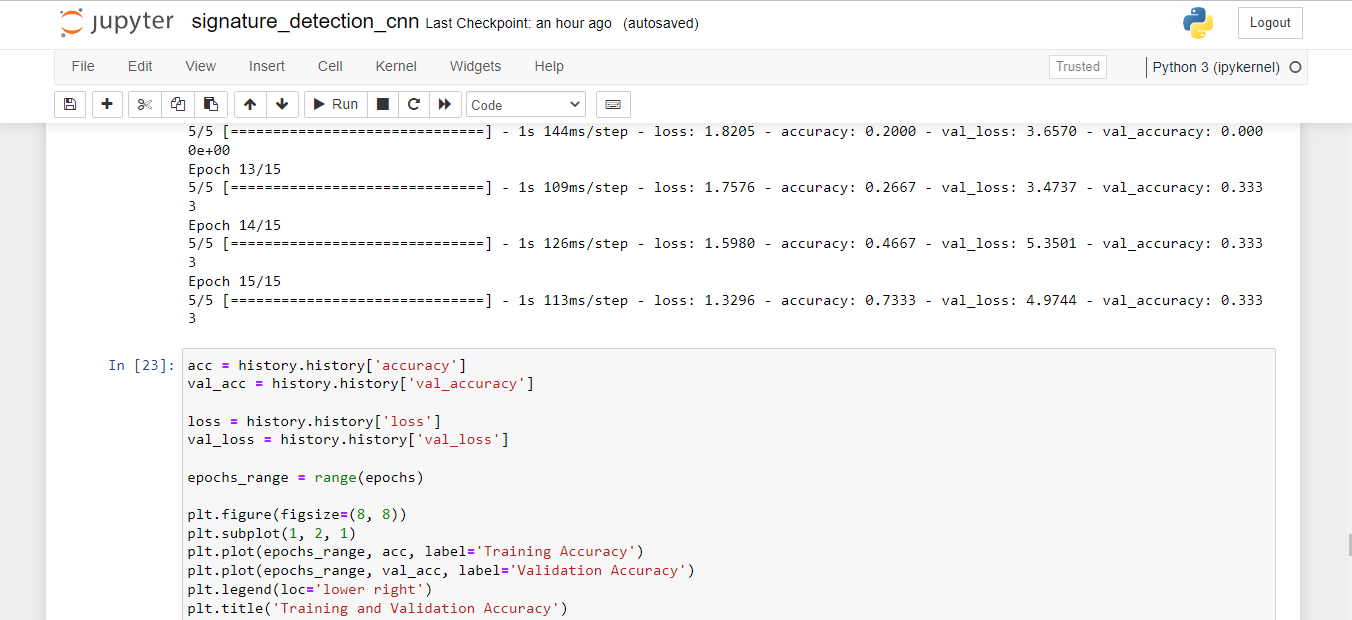




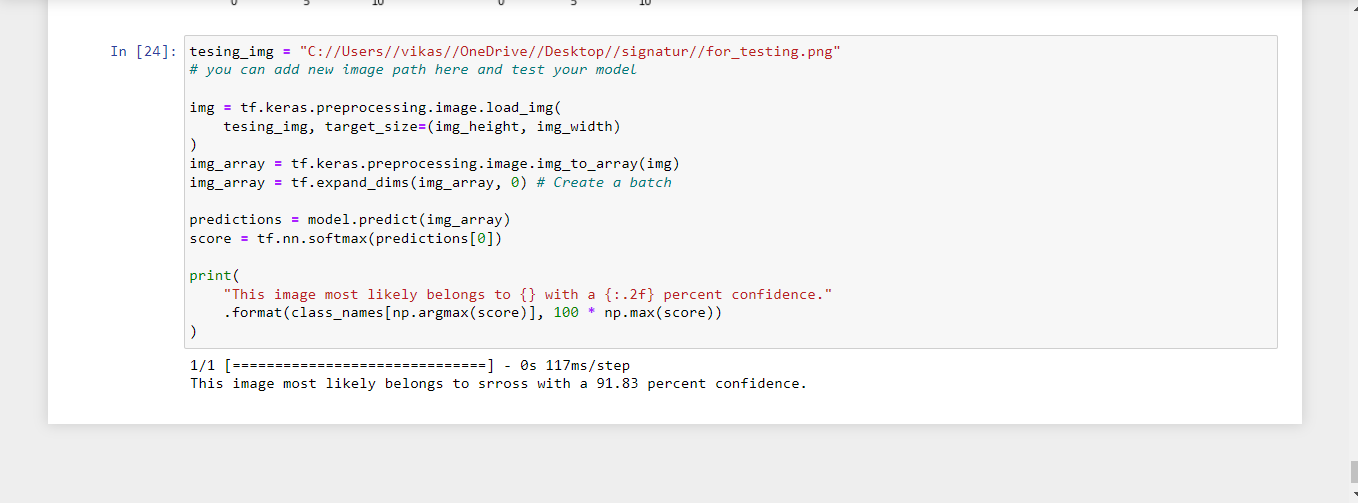








**Final Result You can Predict Your Signature**



**CHALLENGES**

## Challenges faced while adopting Machine Learning

Machine learning is helping organizations make sense of their data, automate business processes, and increase productivity, and gradually profits too. And while companies are keen on adopting machine learning algorithms, they often find themselves struggling to begin the journey.

All the companies are different and their journeys are unique. But essentially, the frequently faced issues in machine learning by companies include common issues like business goals alignment, people’s mindset, and more. Let us discuss and understand the 6 most common issues which companies face during machine learning adoption.

* Inaccessible Data and Data Security. ...
* Infrastructure Requirements for Testing & Experimentation. ...
* Rigid Business Models. ...
* Lack of Talent. ...
* Time-Consuming Implementation. ...
* Affordability.

**LIMITATION**

#### 1. Data Acquisition

Machine Learning requires massive data sets to train on, and these should be inclusive/unbiased, and of good quality. There can also be times where they must wait for new data to be generated.

#### 2. Time and Resources

ML needs enough time to let the algorithms learn and develop enough to fulfill their purpose with a considerable amount of accuracy and relevancy. It also needs massive resources to function. This can mean additional requirements of computer power for you.

#### 3. Interpretation of Results

Another major challenge is the ability to accurately interpret results generated by the algorithms. You must also carefully choose the algorithms for your purpose.

#### 4. High error-susceptibility

[**Machine Learning**](https://en.wikipedia.org/wiki/Machine_learning) is autonomous but highly susceptible to errors. Suppose you train an algorithm with data sets small enough to not be inclusive. You end up with biased predictions coming from a biased training set. This leads to irrelevant advertisements being displayed to customers. In the case of ML, such blunders can set off a chain of errors that can go undetected for long periods of time.

And when they do get noticed, it takes quite some time to recognize the source of the issue, and even longer to correct it.

**CONCLUSION**

Machine Learning can be a Supervised or Unsupervised. If you have lesser amount of data and clearly labelled data for training, opt for Supervised Learning. Unsupervised Learning would generally give better performance and results for large data sets. If you have a huge data set easily available, go for deep learning techniques. You also have learned Reinforcement Learning and

Deep Reinforcement Learning. You now know what Neural Networks are, their applications and limitations. Finally, when it comes to the development of machine learning models of your own, you looked at the choices of various development languages, IDEs and Platforms. Next thing that you need to do is start learning and practicing each machine learning technique.

The subject is vast, it means that there is width, but if you consider the depth, each topic can be learned in a few hours. Each topic is independent of each other. You need to take into consideration one topic at a time, learn it, practice it and implement the algorithm/s in it using a language choice of yours. This is the best way to start studying Machine Learning. Practicing one topic at a time, very soon you would acquire the width that is eventually required of a Machine Learning expert.

**REFERENCES**

 Baldi, P. and Brunak, S. (2002). Bioinformatics: A Machine Learning Approach. Cambridge, MA: MIT Press.   
This book offers a good coverage of machine learning approaches - especially neural networks and hidden Markov models in bioinformatics.

 Baldi, P., Frasconi, P., Smyth, P. (2003). Modeling the Internet and the Web - Probabilistic Methods and Algorithms. New York: Wiley.   
A good introduction to machine learning approaches to text mining and related applications on the web.

 Bishop, C. M. Neural Networks for Pattern Recognition. New York: Oxford University Press (1995).   
This book offers a good coverage of neural networks

 Chakrabarti, S. (2003). Mining the Web, Morgan Kaufmann.

 Cohen, P.R. (1995) Empirical Methods in Artificial Intelligence. Cambridge, MA: MIT Press. This is an excellent reference on experiment design, and hypothesis testing, and related topics that are essential for empirical machine learning research.

 Cowell, R.G., Dawid, A.P., Lauritzen, S.L., and Spiegelhalter,D.J. (1999). Graphical Models and Expert Systems.Berlin: Springer.   
This is a very good introduction to probabilistic graphical models.

 Cristianini, N. and Shawe-Taylor, J. (2000). An Introduction to Support Vector Machines. London: Cambridge University Press.   
This is an excellent introduction to kernel methods for pattern classification.

 Duda, R., Hart, P., and Stork, D. (2001). Pattern Classification. New York: Wiley.   
This is a good text with primary emphasis on statistical methods for pattern classification.

 Hastie, T., Tibshirani, R., and Friedman, J. (2001). The elements of Statistical Learning - Data Mining, Inference, and Prediction. Berlin: Springer-Verlag.   
This is an excellent text that explains some of the key ideas in machine learning within a statistical framework.

 Jordan, M. (2003). Probabilistic Graphical Models. Professor Jordan has kindly shared a pre-publication draft.   
This text has an excellent coverage of generative and discriminative probabilistic models for classification.