codelabs.developers.google.com/codelabs/time-series-forecasting-with-cloud-ai-platform#1

Time Series Forecasting with the Cloud AI Platform and BQML

126 mins remaining | English

1. Overview
2. **Introduction to Time-Series Forecasting**
3. Setup your Notebook environment
4. Explore and Visualize Data
5. Create a Model with BigQuery Time Series Forecasting
6. Build a Custom Forecasting Model
7. Train and Predict in the Cloud
8. Challenge
9. Cleanup

Report a mistake

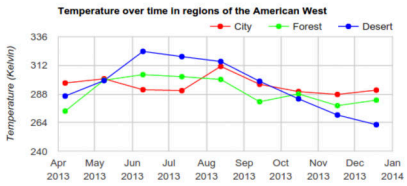
Back

2. Introduction to Time-Series Forecasting

The focus of this codelab is on how to *apply* time-series forecasting techniques using the Google Cloud Platform. It isn't a general time-series forecasting course, but a brief tour of the concepts may be helpful for our users.

Time Series Data

First, what is a time series? It's a dataset with data recorded at regular time intervals. A time-series dataset contains both time and at least one variable that is dependent on time.



Components

A time-series can be decomposed into components:

- **Trend**: moves up or down in a reasonably predictable pattern
- **Seasonal**: repeats over a specific period such as a day, week, month, season, etc.
- **Random**: residual fluctuations

There can be multiple layers of seasonality. For example, a call center might see a pattern in call volume on certain days of the week as well as on given months. The residual might be able to be explained by other variables besides

Next

1e7

codelabs.developers.google.com/codelabs/time-series-forecasting-with-cloud-ai-platform#2

Time Series Forecasting with the Cloud AI Platform and BQML

121 mins remaining | English

1. Overview
2. Introduction to Time-Series Forecasting
3. **Setup your Notebook environment**
4. Explore and Visualize Data
5. Create a Model with BigQuery Time Series Forecasting
6. Build a Custom Forecasting Model
7. Train and Predict in the Cloud
8. Challenge
9. Cleanup

Report a mistake

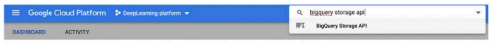
Back

3. Setup your Notebook environment

Now that we have gone through a brief introduction to the data, let's now set up our model development environment.

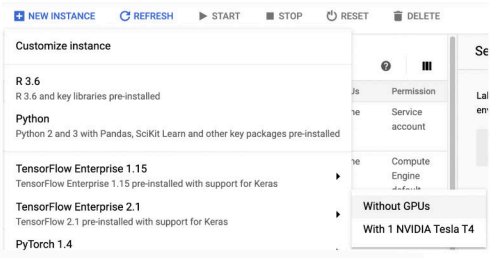
Step 1: Enable APIs

The BigQuery connector uses the BigQuery Storage API. Search for the **BigQuery Storage API** in the console and enable the API if it is currently disabled.



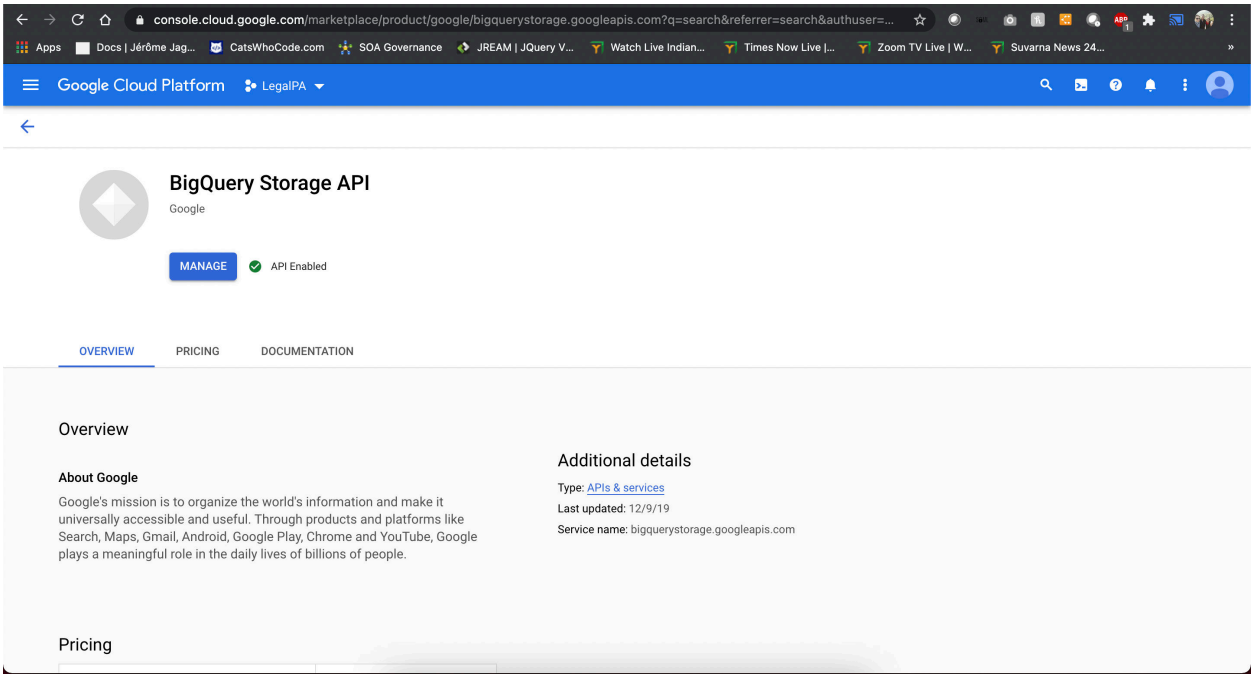
Step 2: Create an AI Platform Notebooks instance

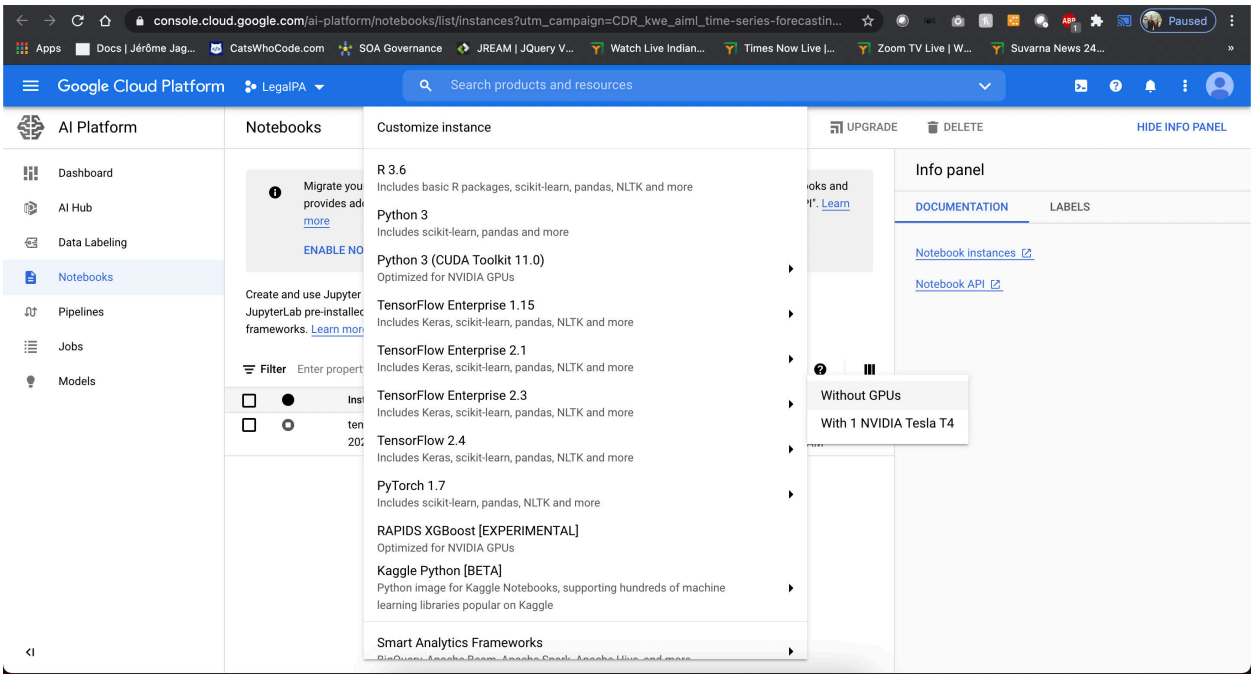
Navigate to AI Platform Notebooks section of your Cloud Console and click **New Instance**. Then select the latest **TensorFlow Enterprise 2.x** instance type **without GPUs**:

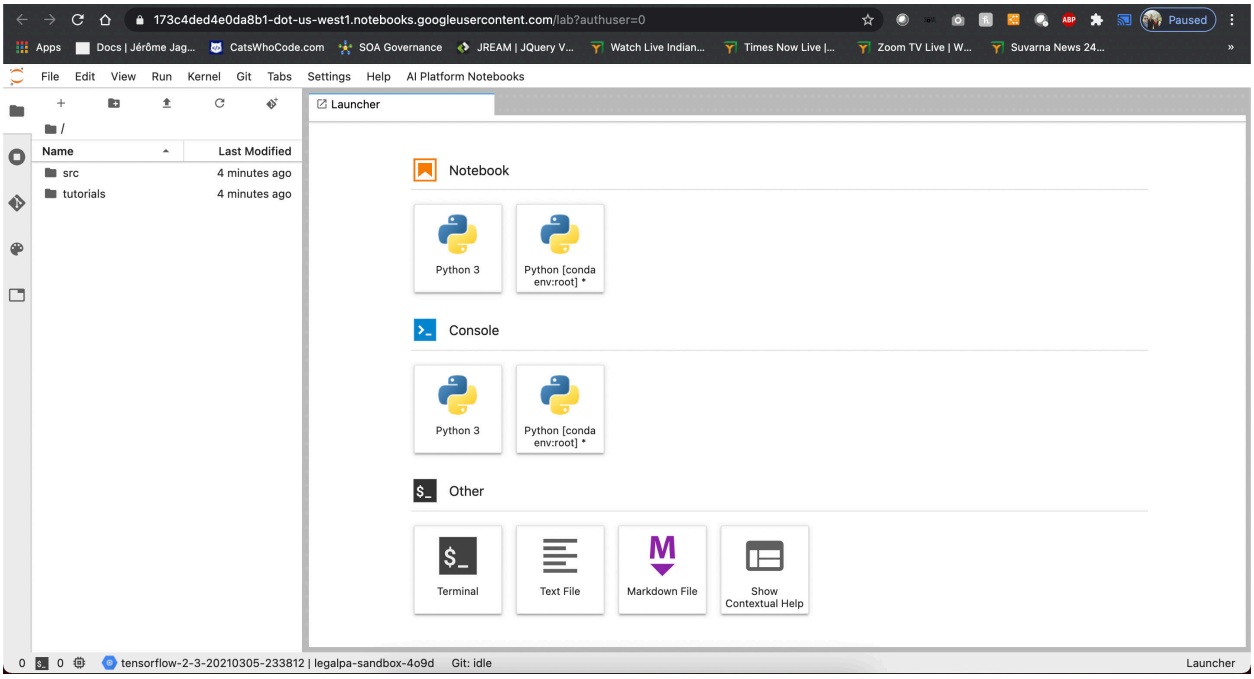


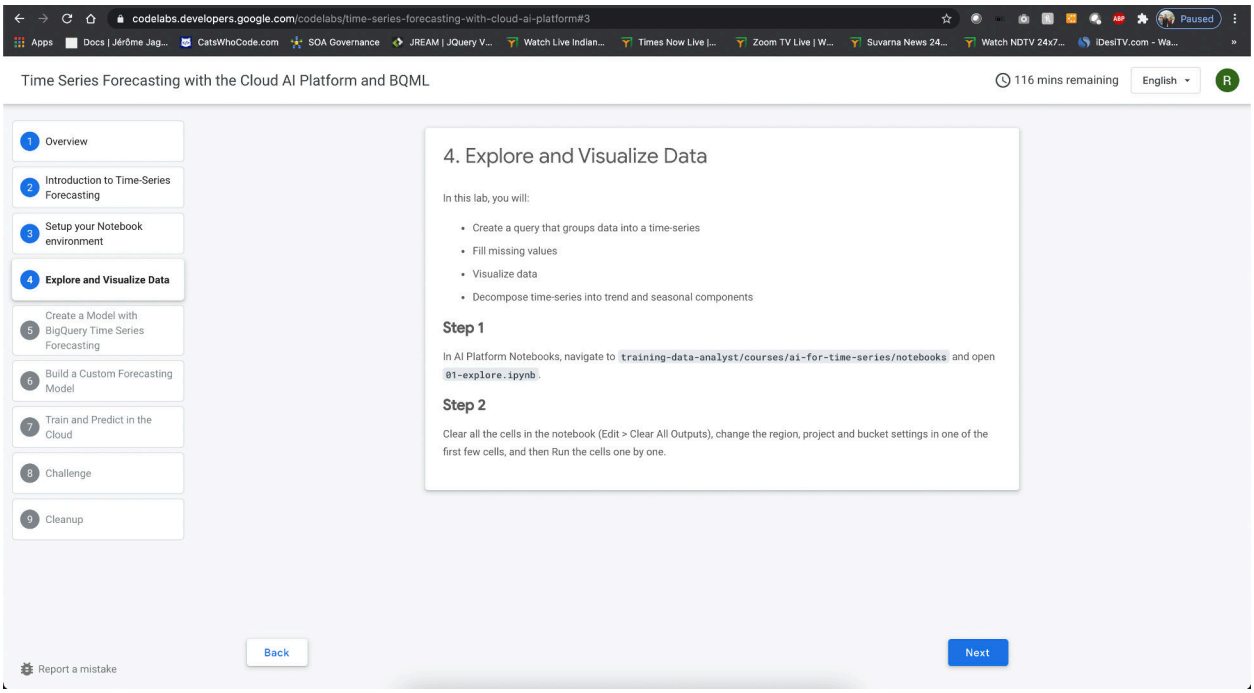
Next

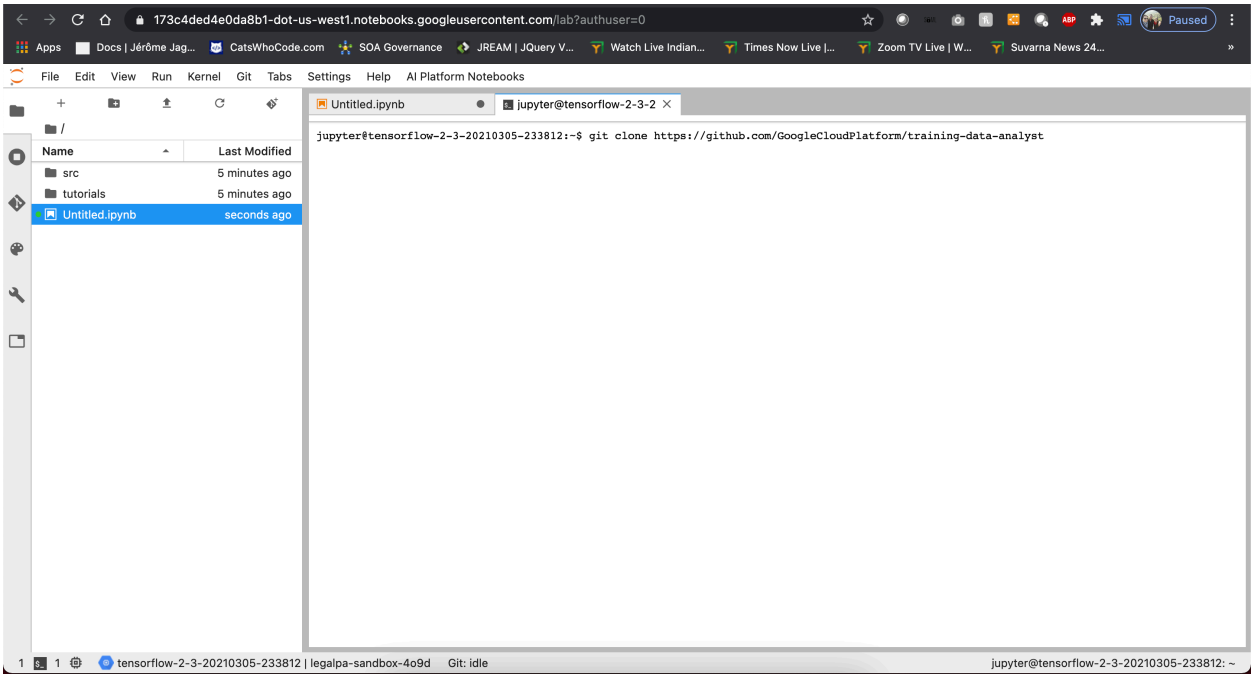
Use the default options and then click **Create**. Once the instance has been created, select **Open JupyterLab**:

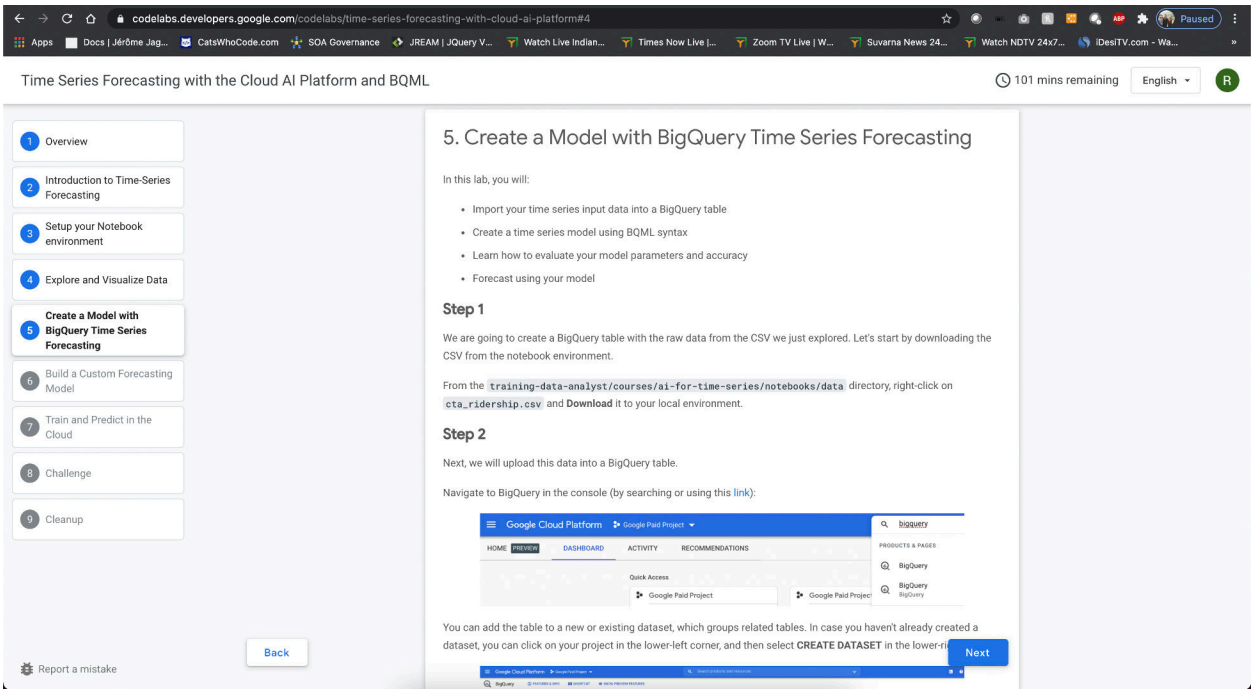

console.cloud.google.com/marketplace/product/google/bigquerystorage.googleapis.com?q=search&referrer=search&authuser=...
Google Cloud Platform
LegalPA
BigQuery Storage API
Google
MANAGE
API Enabled
OVERVIEW
PRICING
DOCUMENTATION
Overview
About Google
Google's mission is to organize the world's information and make it universally accessible and useful. Through products and platforms like Search, Maps, Gmail, Android, Google Play, Chrome and YouTube, Google plays a meaningful role in the daily lives of billions of people.
Additional details
Type: APIs & services
Last updated: 12/9/19
Service name: bigquerystorage.googleapis.com
Pricing

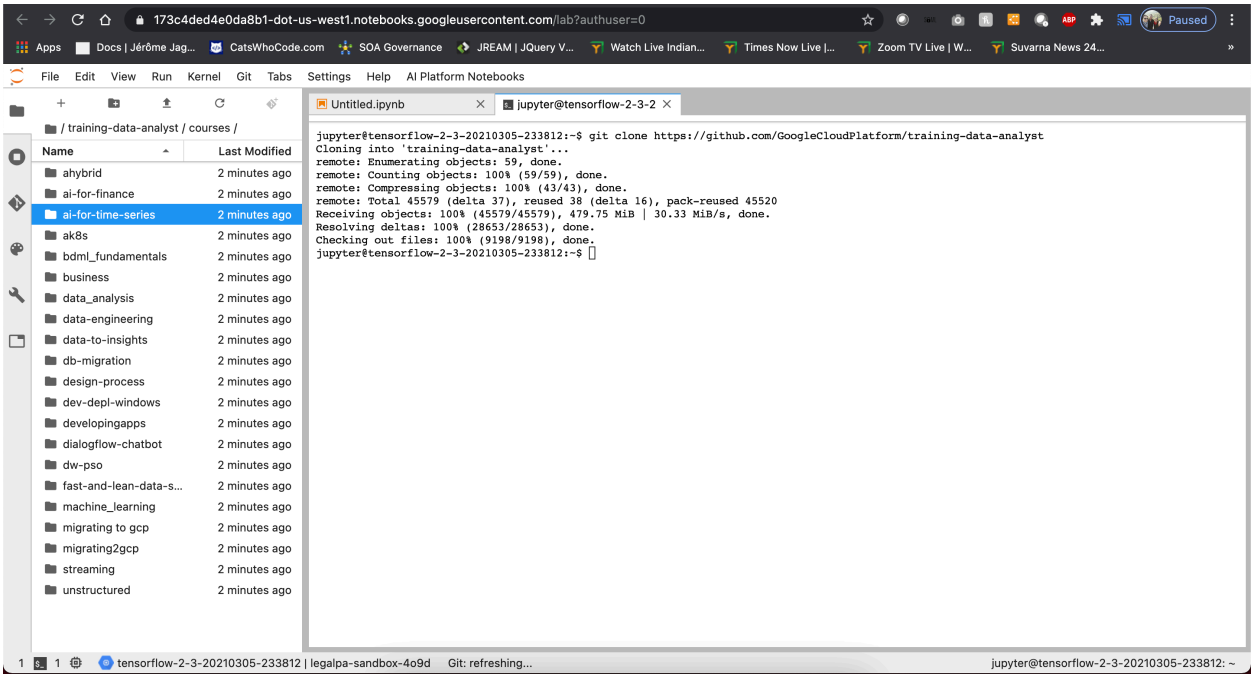

console.cloud.google.com/ai-platform/notebooks/list/instances?utm_campaign=CDR_kwe_aiml_time-series-forecastin...
Google Cloud Platform
LegalPA
Search products and resources
AI Platform
Notebooks
Dashboard
AI Hub
Data Labeling
Notebooks
Pipelines
Jobs
Models
Customize instance
R 3.6
Includes basic R packages, scikit-learn, pandas, NLTK and more
Python 3
Includes scikit-learn, pandas and more
Python 3 (CUDA Toolkit 11.0)
Optimized for NVIDIA GPUs
TensorFlow Enterprise 1.15
Includes Keras, scikit-learn, pandas, NLTK and more
TensorFlow Enterprise 2.1
Includes Keras, scikit-learn, pandas, NLTK and more
TensorFlow Enterprise 2.3
Includes Keras, scikit-learn, pandas, NLTK and more
TensorFlow 2.4
Includes Keras, scikit-learn, pandas, NLTK and more
PyTorch 1.7
Includes scikit-learn, pandas, NLTK and more
RAPIDS XGBoost [EXPERIMENTAL]
Optimized for NVIDIA GPUs
Kaggle Python [BETA]
Python image for Kaggle Notebooks, supporting hundreds of machine learning libraries popular on Kaggle
Smart Analytics Frameworks
Without GPUs
With 1 NVIDIA Tesla T4
UPGRADE
DELETE
HIDE INFO PANEL
Info panel
DOCUMENTATION
LABELS
Notebook instances
Notebook API
Filter

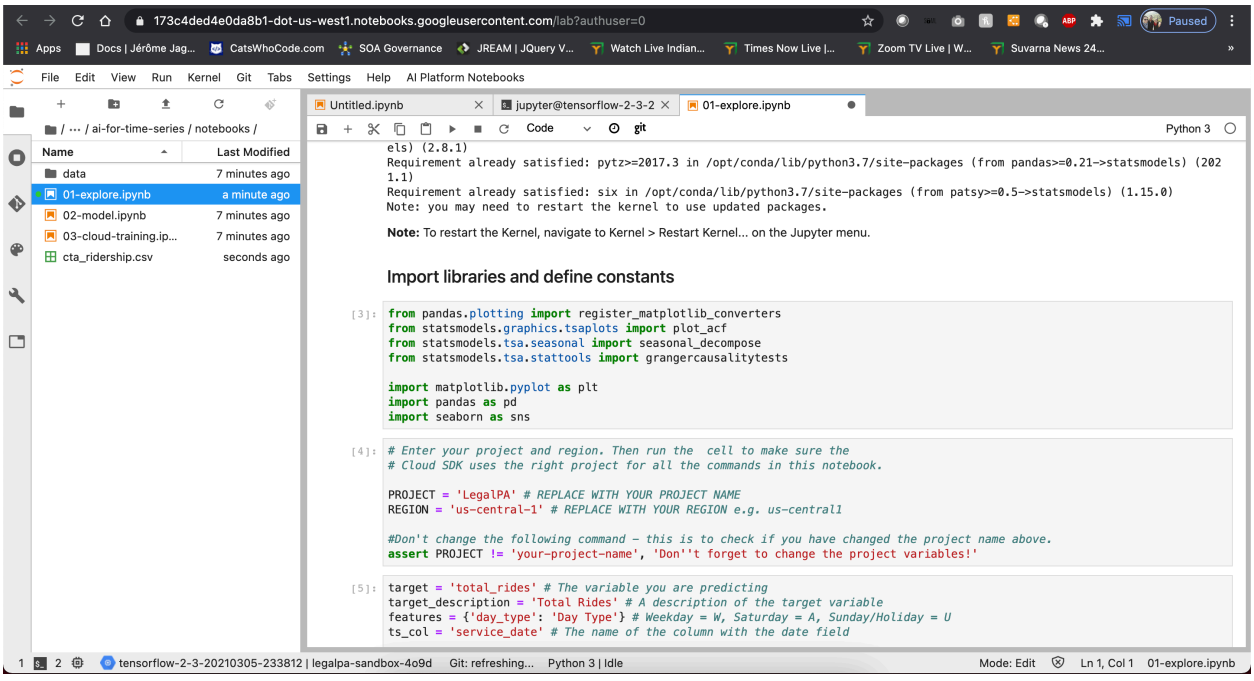

173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0
Paused
Apps
Docs | Jérôme Jag...
CatsWhoCode.com
SOA Governance
JREAM | JQuery V...
Watch Live Indian...
Times Now Live |...
Zoom TV Live | W...
Suvarna News 24...
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
Launcher
/
Name
Last Modified
src
4 minutes ago
tutorials
4 minutes ago
Notebook
Python 3
Python [conda env:root] *
Console
Python 3
Python [conda env:root] *
Other
Terminal
Text File
Markdown File
Show Contextual Help
0 0 tensorflow-2-3-20210305-233812 | legalpa-sandbox-4o9d Git: idle
Launcher

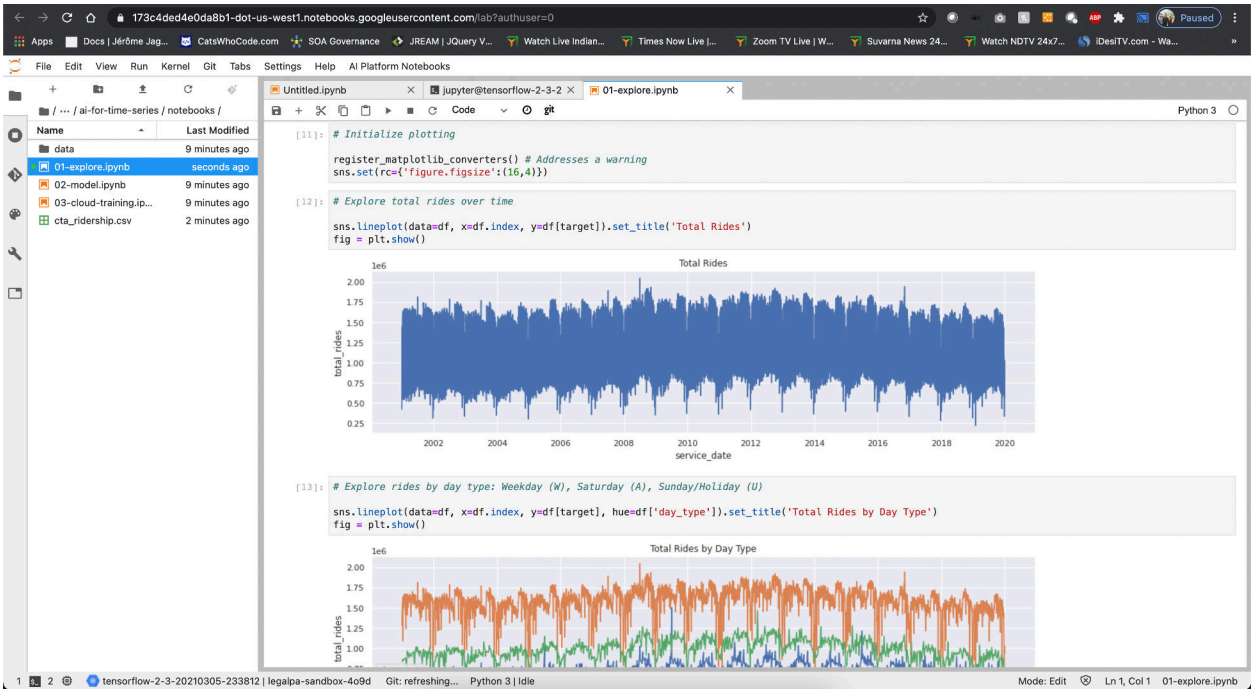

codelabs.developers.google.com/codelabs/time-series-forecasting-with-cloud-ai-platform#3
Paused
Time Series Forecasting with the Cloud AI Platform and BQML
116 mins remaining
English
1 Overview
2 Introduction to Time-Series Forecasting
3 Setup your Notebook environment
4 Explore and Visualize Data
5 Create a Model with BigQuery Time Series Forecasting
6 Build a Custom Forecasting Model
7 Train and Predict in the Cloud
8 Challenge
9 Cleanup
4. Explore and Visualize Data
In this lab, you will:
Create a query that groups data into a time-series
Fill missing values
Visualize data
Decompose time-series into trend and seasonal components
Step 1
In AI Platform Notebooks, navigate to training-data-analyst/courses/ai-for-time-series/notebooks and open 01-explore.ipynb.
Step 2
Clear all the cells in the notebook (Edit > Clear All Outputs), change the region, project and bucket settings in one of the first few cells, and then Run the cells one by one.
Report a mistake
Back
Next

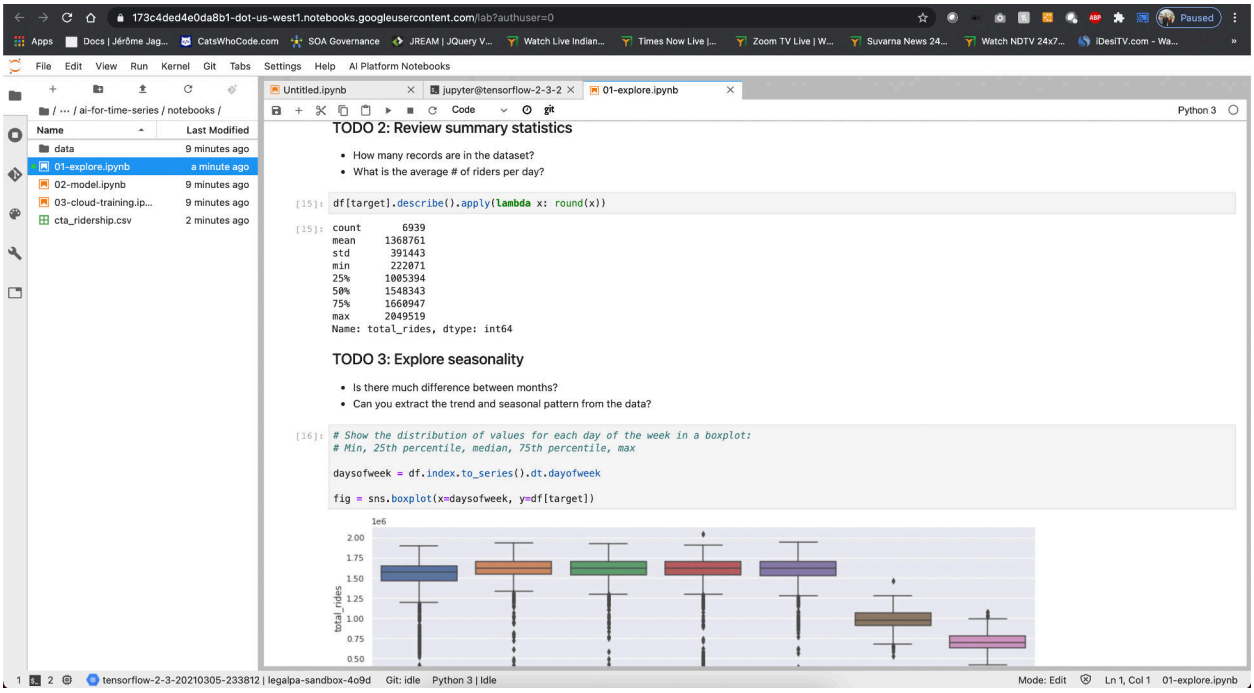

173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0
Paused
Apps
Docs | Jérôme Jag...
CatsWhoCode.com
SOA Governance
JREAM | JQuery V...
Watch Live Indian...
Times Now Live |...
Zoom TV Live | W...
Suvarna News 24...
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
Untitled.ipynb
jupyter@tensorflow-2-3-2
jupyter@tensorflow-2-3-20210305-233812:~$ git clone https://github.com/GoogleCloudPlatform/training-data-analyst
Name
Last Modified
src
5 minutes ago
tutorials
5 minutes ago
Untitled.ipynb
seconds ago
tensorflow-2-3-20210305-233812 | legalpa-sandbox-4o9d
Git: idle
jupyter@tensorflow-2-3-20210305-233812: ~

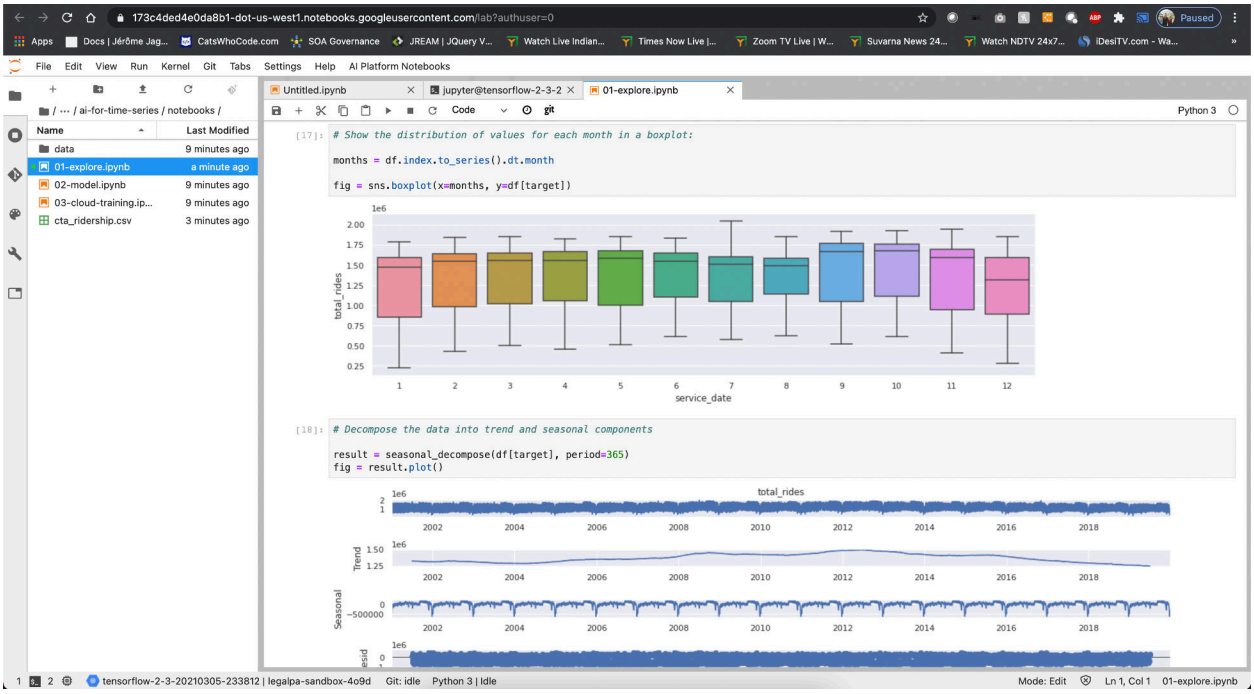

codelabs.developers.google.com/codelabs/time-series-forecasting-with-cloud-ai-platform#4
Paused
Time Series Forecasting with the Cloud AI Platform and BQML
101 mins remaining
English
1 Overview
2 Introduction to Time-Series Forecasting
3 Setup your Notebook environment
4 Explore and Visualize Data
5 Create a Model with BigQuery Time Series Forecasting
6 Build a Custom Forecasting Model
7 Train and Predict in the Cloud
8 Challenge
9 Cleanup
Report a mistake
Back
Next
5. Create a Model with BigQuery Time Series Forecasting
In this lab, you will:
Import your time series input data into a BigQuery table
Create a time series model using BQML syntax
Learn how to evaluate your model parameters and accuracy
Forecast using your model
Step 1
We are going to create a BigQuery table with the raw data from the CSV we just explored. Let's start by downloading the CSV from the notebook environment.
From the training-data-analyst/courses/ai-for-time-series/notebooks/data directory, right-click on cta_ridership.csv and Download it to your local environment.
Step 2
Next, we will upload this data into a BigQuery table.
Navigate to BigQuery in the console (by searching or using this link):
Google Cloud Platform
Google Paid Project
bigquery
HOME
DASHBOARD
ACTIVITY
RECOMMENDATIONS
Quick Access
BigQuery
You can add the table to a new or existing dataset, which groups related tables. In case you haven't already created a dataset, you can click on your project in the lower-left corner, and then select CREATE DATASET in the lower-r

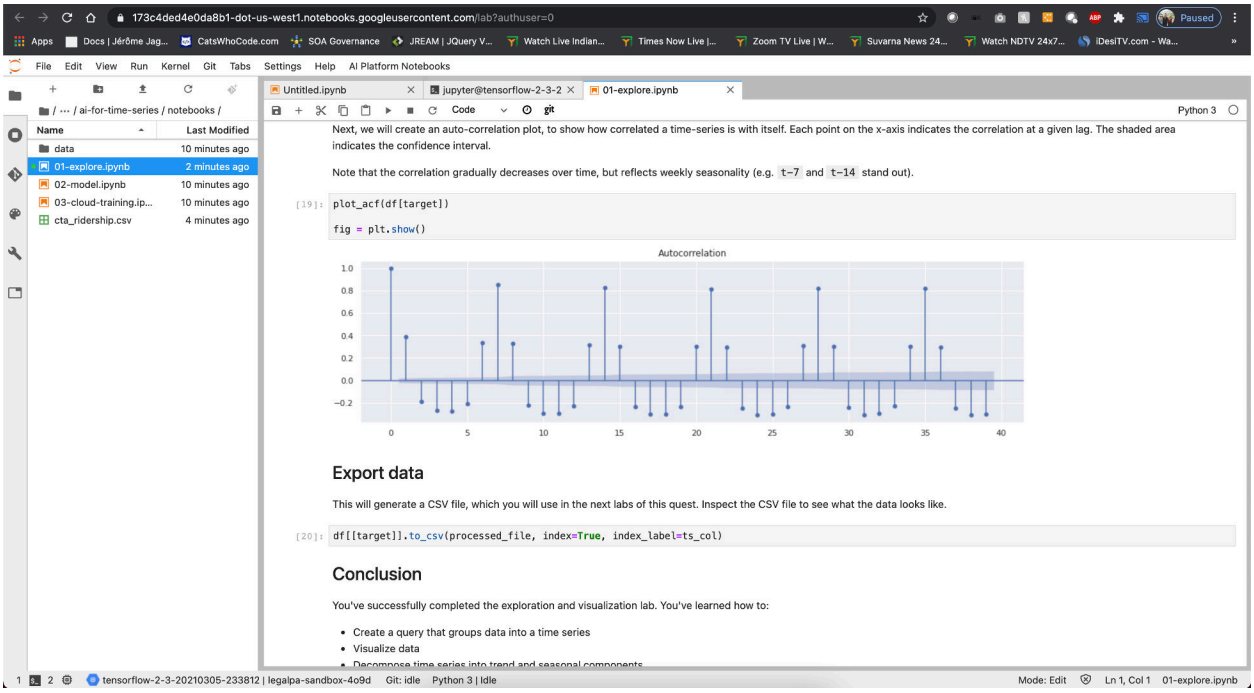

173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0
Paused
Apps
Docs | Jérôme Jag...
CatsWhoCode.com
SOA Governance
JREAM | JQuery V...
Watch Live Indian...
Times Now Live |...
Zoom TV Live | W...
Suvarna News 24...
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
/ training-data-analyst / courses /
Name
Last Modified
ahybrid 2 minutes ago
ai-for-finance 2 minutes ago
ai-for-time-series 2 minutes ago
ak8s 2 minutes ago
bdml_fundamentals 2 minutes ago
business 2 minutes ago
data_analysis 2 minutes ago
data-engineering 2 minutes ago
data-to-insights 2 minutes ago
db-migration 2 minutes ago
design-process 2 minutes ago
dev-depl-windows 2 minutes ago
developingapps 2 minutes ago
dialogflow-chatbot 2 minutes ago
dw-pso 2 minutes ago
fast-and-lean-data-s... 2 minutes ago
machine_learning 2 minutes ago
migrating to gcp 2 minutes ago
migrating2gcp 2 minutes ago
streaming 2 minutes ago
unstructured 2 minutes ago
Untitled.ipynb
jupyter@tensorflow-2-3-2
jupyter@tensorflow-2-3-20210305-233812:~$ git clone https://github.com/GoogleCloudPlatform/training-data-analyst
Cloning into 'training-data-analyst'...
remote: Enumerating objects: 59, done.
remote: Counting objects: 100% (59/59), done.
remote: Compressing objects: 100% (43/43), done.
remote: Total 45579 (delta 37), reused 38 (delta 16), pack-reused 45520
Receiving objects: 100% (45579/45579), 479.75 MiB | 30.33 MiB/s, done.
Resolving deltas: 100% (28653/28653), done.
Checking out files: 100% (9198/9198), done.
jupyter@tensorflow-2-3-20210305-233812:~$
1 1 tensorflow-2-3-20210305-233812 | legalpa-sandbox-4o9d Git: refreshing... jupyter@tensorflow-2-3-20210305-233812: ~


173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0
Paused
Apps
Docs | Jérôme Jag...
CatsWhoCode.com
SOA Governance
JREAM | JQuery V...
Watch Live Indian...
Times Now Live |...
Zoom TV Live | W...
Suvarna News 24...
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
/ ... / ai-for-time-series / notebooks /
Name
Last Modified
data 7 minutes ago
01-explore.ipynb a minute ago
02-model.ipynb 7 minutes ago
03-cloud-training.ip... 7 minutes ago
cta_ridership.csv seconds ago
Untitled.ipynb
jupyter@tensorflow-2-3-2
01-explore.ipynb
Code
git
Python 3
els) (2.8.1)
Requirement already satisfied: pytz>=2017.3 in /opt/conda/lib/python3.7/site-packages (from pandas>=0.21->statsmodels) (2021.1)
Requirement already satisfied: six in /opt/conda/lib/python3.7/site-packages (from patsy>=0.5->statsmodels) (1.15.0)
Note: you may need to restart the kernel to use updated packages.
Note: To restart the Kernel, navigate to Kernel > Restart Kernel... on the Jupyter menu.
Import libraries and define constants
[3]:
from pandas.plotting import register_matplotlib_converters
from statsmodels.graphics.tsaplots import plot_acf
from statsmodels.tsa.seasonal import seasonal_decompose
from statsmodels.tsa.stattools import grangercausalitytests
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
[4]:
# Enter your project and region. Then run the cell to make sure the
# Cloud SDK uses the right project for all the commands in this notebook.
PROJECT = 'LegalPA' # REPLACE WITH YOUR PROJECT NAME
REGION = 'us-central1' # REPLACE WITH YOUR REGION e.g. us-central1
#Don't change the following command - this is to check if you have changed the project name above.
assert PROJECT != 'your-project-name', 'Don''t forget to change the project variables!'
[5]:
target = 'total_rides' # The variable you are predicting
target_description = 'Total Rides' # A description of the target variable
features = {'day_type': 'Day Type'} # Weekday = W, Saturday = A, Sunday/Holiday = U
ts_col = 'service_date' # The name of the column with the date field
1 2 tensorflow-2-3-20210305-233812 | legalpa-sandbox-4o9d Git: refreshing... Python 3 | Idle Mode: Edit Ln 1, Col 1 01-explore.ipynb

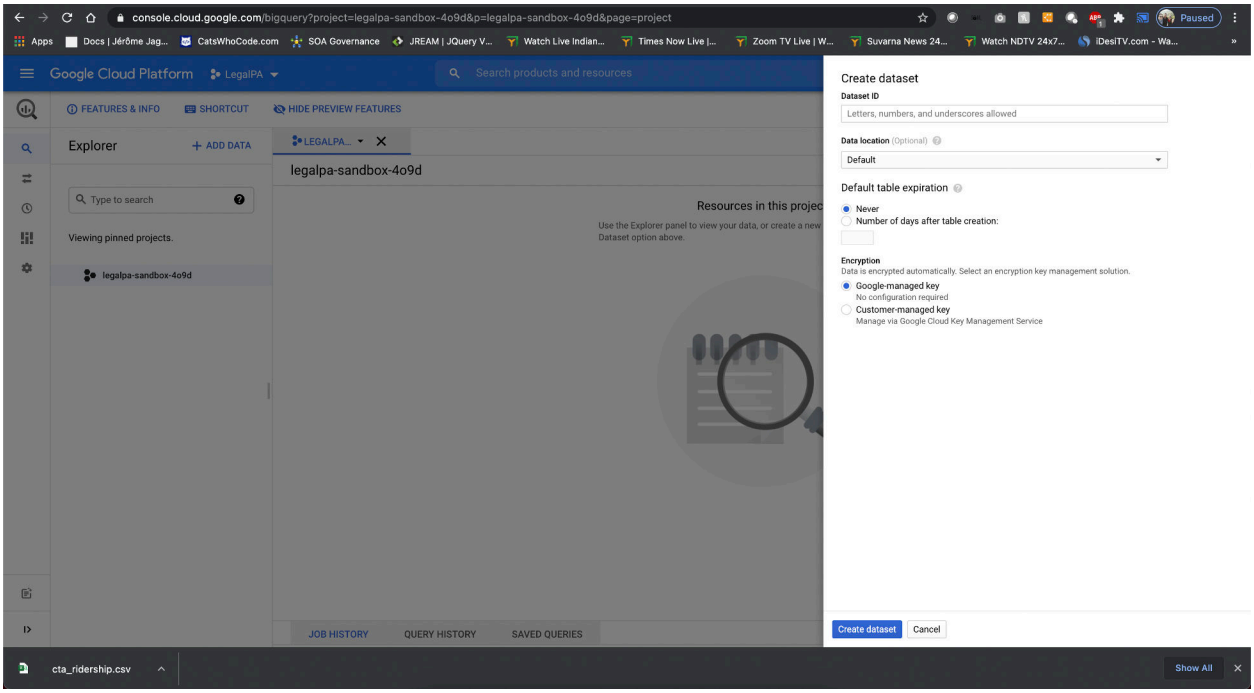

173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0
Paused
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
/ ··· / ai-for-time-series / notebooks /
Name
Last Modified
data 9 minutes ago
01-explore.ipynb seconds ago
02-model.ipynb 9 minutes ago
03-cloud-training.ip... 9 minutes ago
cta_ridership.csv 2 minutes ago
Untitled.ipynb
jupyter@tensorflow-2-3-2
01-explore.ipynb
Code
Python 3
[11]:
Initialize plotting
register_matplotlib_converters() # Addresses a warning
sns.set(rc={'figure.figsize':(16,4)})
[12]:
# Explore total rides over time
sns.lineplot(data=df, x=df.index, y=df[target]).set_title('Total Rides')
fig = plt.show()
Total Rides
1e6
total_rides
service_date
[13]:
# Explore rides by day type: Weekday (W), Saturday (A), Sunday/Holiday (U)
sns.lineplot(data=df, x=df.index, y=df[target], hue=df['day_type']).set_title('Total Rides by Day Type')
fig = plt.show()
Total Rides by Day Type
tensorflow-2-3-20210305-233812 | legalpa-sandbox-4o9d
Git: refreshing...
Python 3 | Idle
Mode: Edit
Ln 1, Col 1
01-explore.ipynb


173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0
Paused
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
/ ··· / ai-for-time-series / notebooks /
Name
Last Modified
data 9 minutes ago
01-explore.ipynb a minute ago
02-model.ipynb 9 minutes ago
03-cloud-training.ip... 9 minutes ago
cta_ridership.csv 2 minutes ago
Untitled.ipynb
jupyter@tensorflow-2-3-2
01-explore.ipynb
Code
Python 3
TODO 2: Review summary statistics
How many records are in the dataset?
What is the average # of riders per day?
[15]:
df[target].describe().apply(lambda x: round(x))
[15]:
count 6939
mean 1368761
std 391443
min 222071
25% 1005394
50% 1548343
75% 1660947
max 2049519
Name: total_rides, dtype: int64
TODO 3: Explore seasonality
Is there much difference between months?
Can you extract the trend and seasonal pattern from the data?
[16]:
# Show the distribution of values for each day of the week in a boxplot:
# Min, 25th percentile, median, 75th percentile, max
daysofweek = df.index.to_series().dt.dayofweek
fig = sns.boxplot(x=daysofweek, y=df[target])
1e6
total_rides
tensorflow-2-3-20210305-233812 | legalpa-sandbox-4o9d
Git: idle
Python 3 | Idle
Mode: Edit
Ln 1, Col 1
01-explore.ipynb

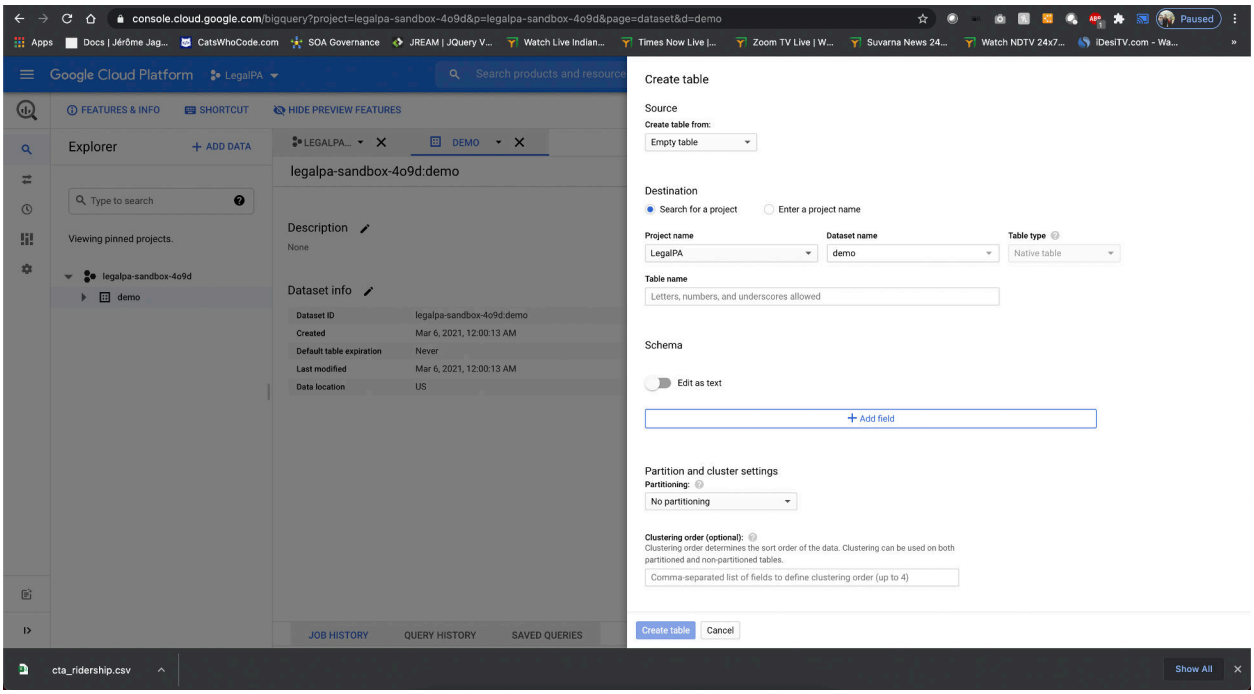


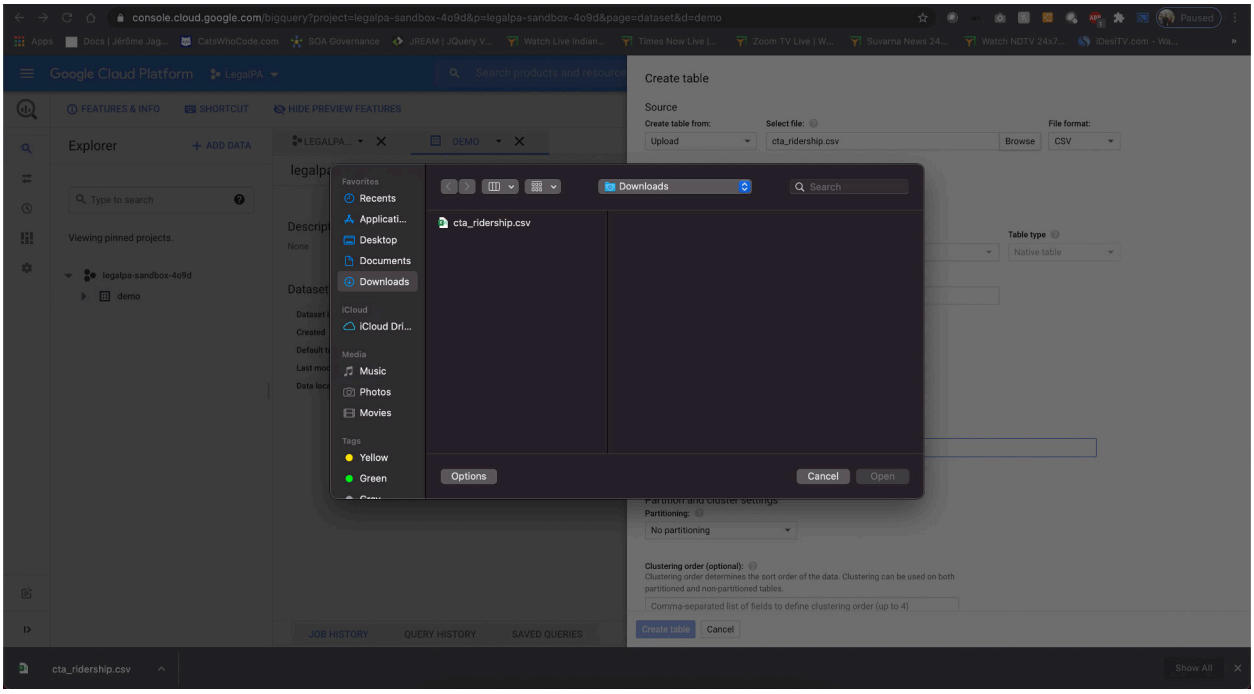


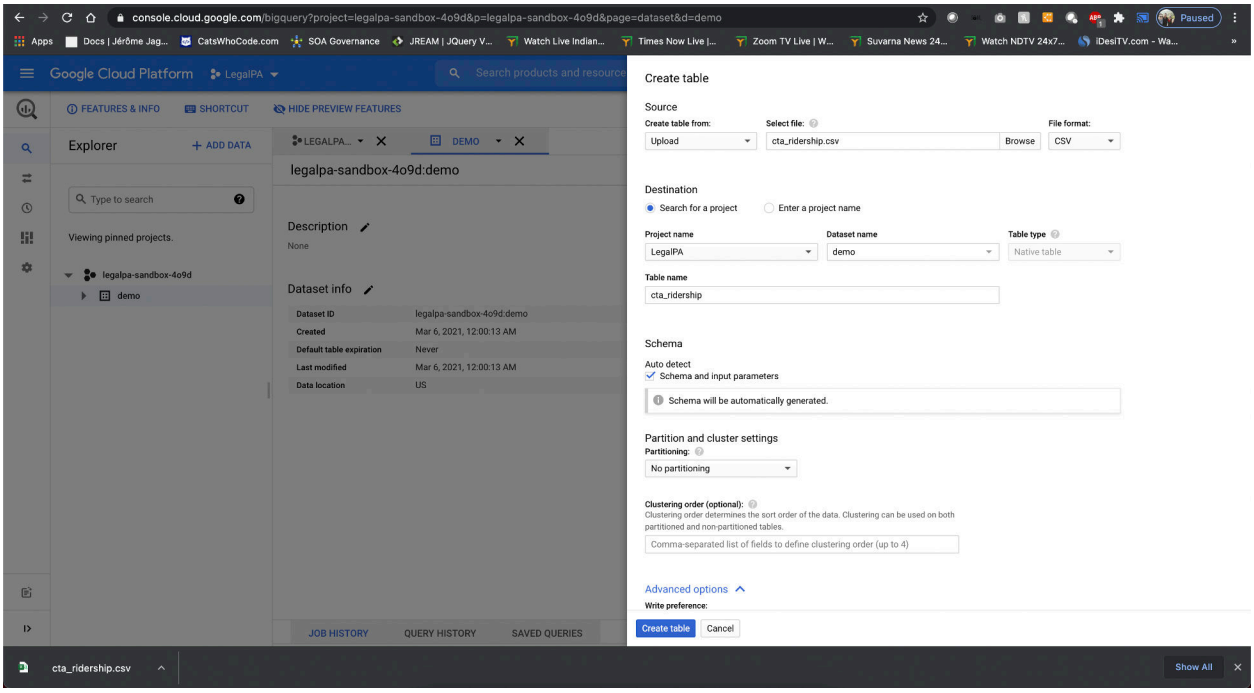
Creating the data set

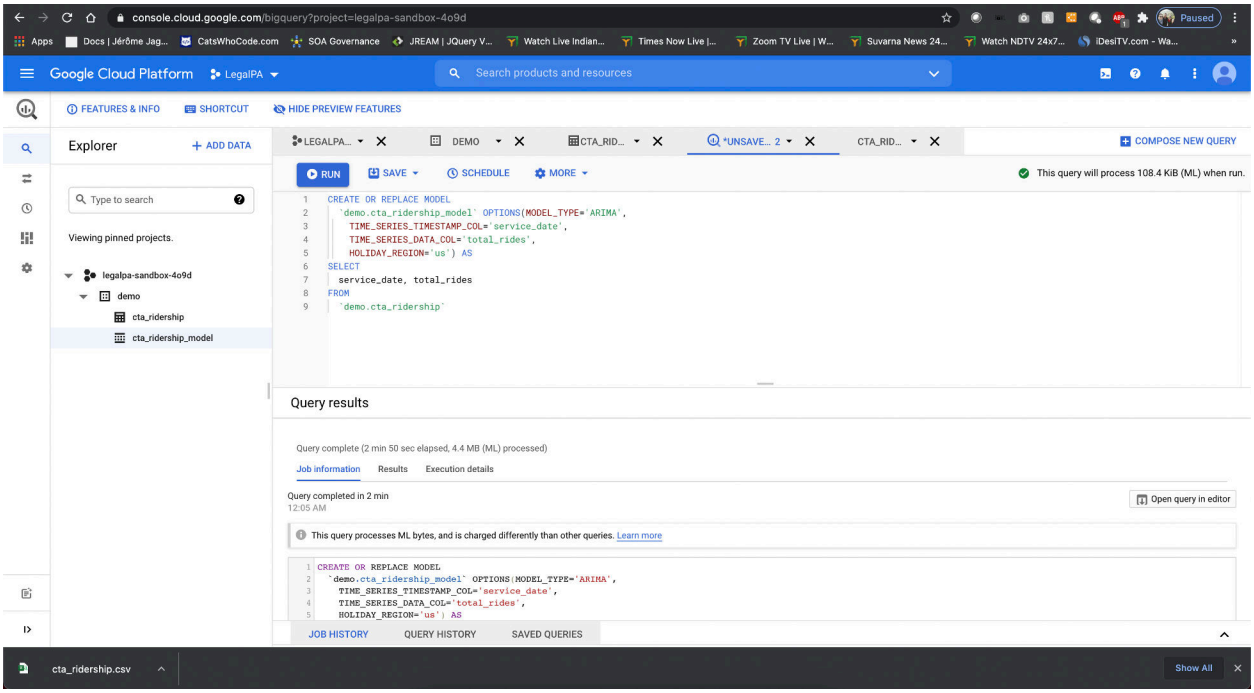


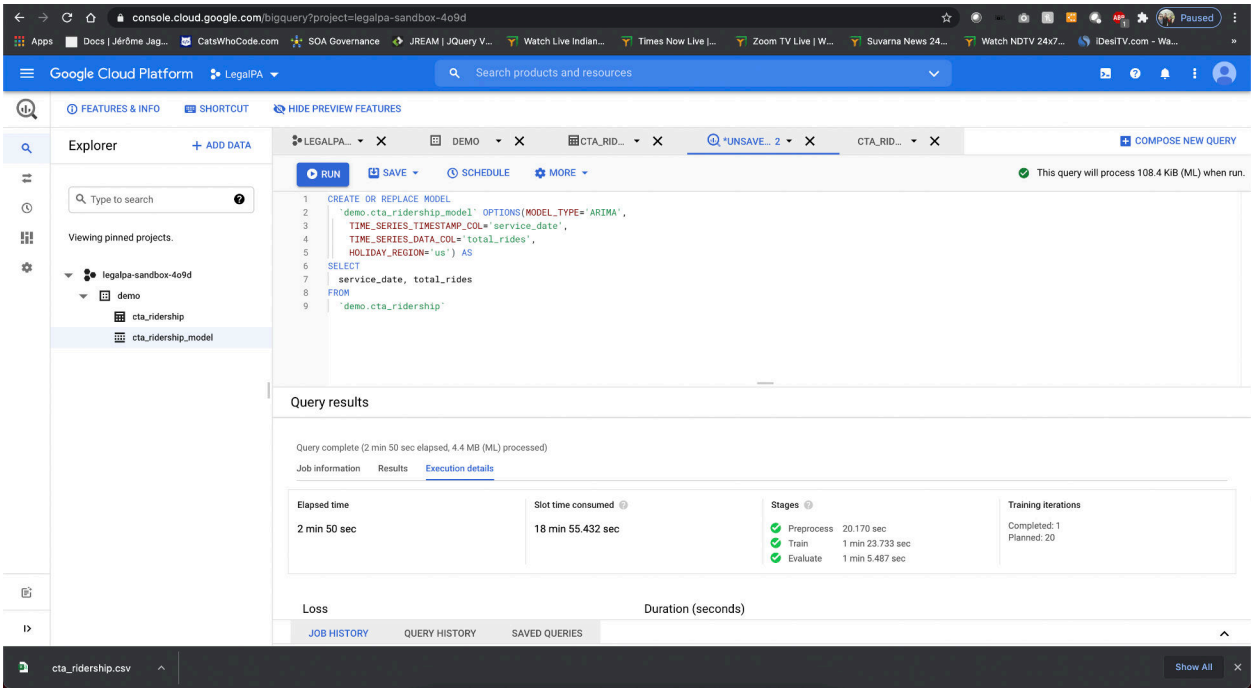
Create a new table

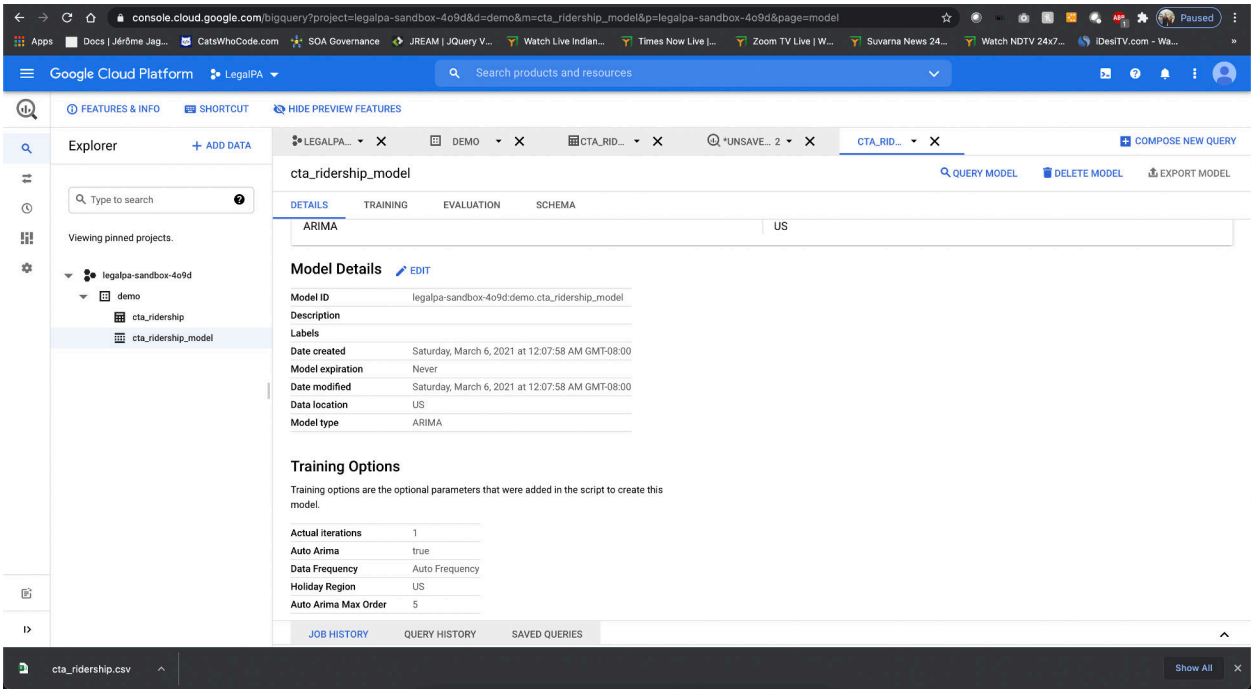



console.cloud.google.com/bigquery?project=legalpa-sandbox-4o9d&p=legalpa-sandbox-4o9d&page=dataset&d=demo
Google Cloud Platform
LegalPA
Explorer
ADD DATA
Type to search
Viewing pinned projects.
legalpa-sandbox-4o9d
demo
Create table
Source
Create table from:
Upload
Select file:
cta_ridership.csv
Browse
File format:
CSV
Table type
Native table
Partitioning:
No partitioning
Clustering order (optional):
Clustering order determines the sort order of the data. Clustering can be used on both partitioned and non-partitioned tables.
Comma-separated list of fields to define clustering order (up to 4)
Create table
Cancel
Downloads
Search
Favorites
Recents
Desktop
Documents
Downloads
iCloud
Media
Music
Photos
Movies
Tags
Yellow
Green
Options
Cancel
Open
JOB HISTORY
QUERY HISTORY
SAVED QUERIES
cta_ridership.csv
Show All

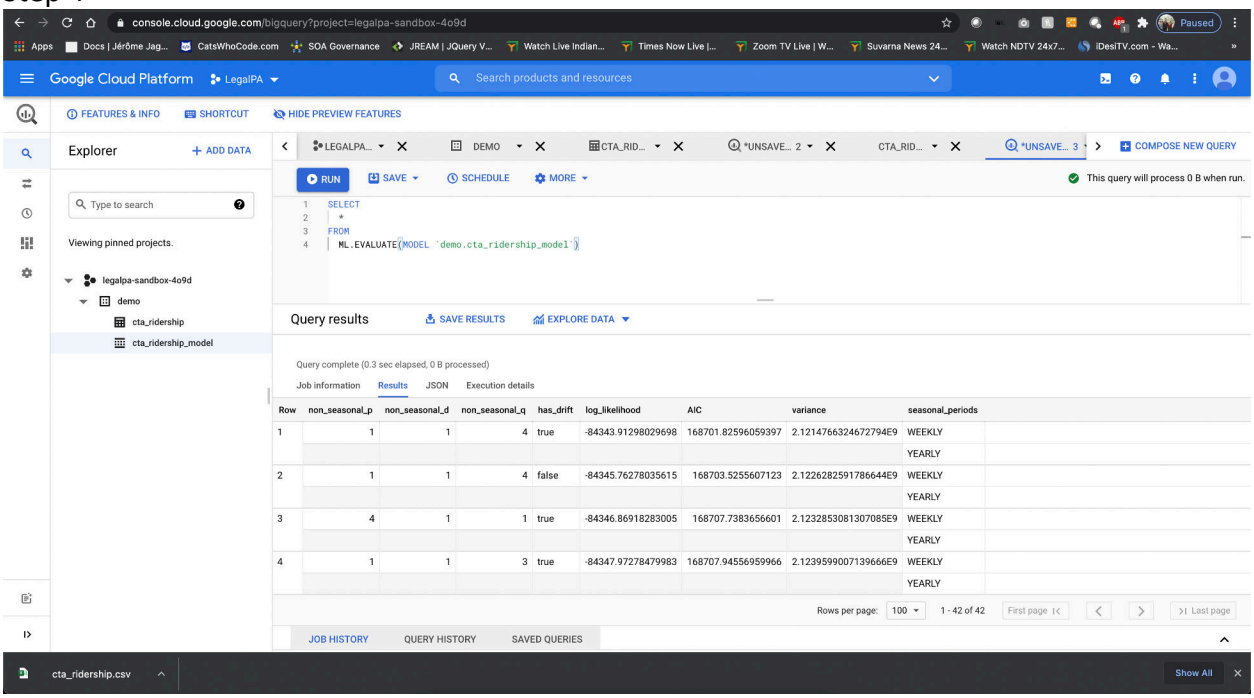

console.cloud.google.com/bigquery?project=legalpa-sandbox-4o9d&p=legalpa-sandbox-4o9d&page=dataset&d=demo
Google Cloud Platform
LegalPA
Explorer
ADD DATA
Type to search
Viewing pinned projects.
legalpa-sandbox-4o9d
demo
legalpa-sandbox-4o9d:demo
Description
None
Dataset info
Dataset ID
legalpa-sandbox-4o9d:demo
Created
Mar 6, 2021, 12:00:13 AM
Default table expiration
Never
Last modified
Mar 6, 2021, 12:00:13 AM
Data location
US
Create table
Source
Create table from:
Upload
Select file:
cta_ridership.csv
Browse
File format:
CSV
Destination
Search for a project
Enter a project name
Project name
LegalPA
Dataset name
demo
Table type
Native table
Table name
cta_ridership
Schema
Auto detect
Schema and input parameters
Schema will be automatically generated.
Partition and cluster settings
Partitioning:
No partitioning
Clustering order (optional):
Clustering order determines the sort order of the data. Clustering can be used on both partitioned and non-partitioned tables.
Comma-separated list of fields to define clustering order (up to 4)
Advanced options
Write preference:
Create table
Cancel
JOB HISTORY
QUERY HISTORY
SAVED QUERIES
cta_ridership.csv
Show All


console.cloud.google.com/bigquery?project=legalpa-sandbox-4o9d
Google Cloud Platform
LegalPA
Search products and resources
FEATURES & INFO
SHORTCUT
HIDE PREVIEW FEATURES
Explorer
ADD DATA
Type to search
Viewing pinned projects.
legalpa-sandbox-4o9d
demo
cta_ridership
cta_ridership_model
COMPOSE NEW QUERY
RUN
SAVE
SCHEDULE
MORE
This query will process 108.4 KiB (ML) when run.
CREATE OR REPLACE MODEL
`demo.cta_ridership_model` OPTIONS(MODEL_TYPE='ARIMA',
TIME_SERIES_TIMESTAMP_COL='service_date',
TIME_SERIES_DATA_COL='total_rides',
HOLIDAY_REGION='us') AS
SELECT
service_date, total_rides
FROM
`demo.cta_ridership`
Query results
Query complete (2 min 50 sec elapsed, 4.4 MB (ML) processed)
Job information
Results
Execution details
Query completed in 2 min
12:05 AM
Open query in editor
This query processes ML bytes, and is charged differently than other queries. Learn more
JOB HISTORY
QUERY HISTORY
SAVED QUERIES
cta_ridership.csv
Show All

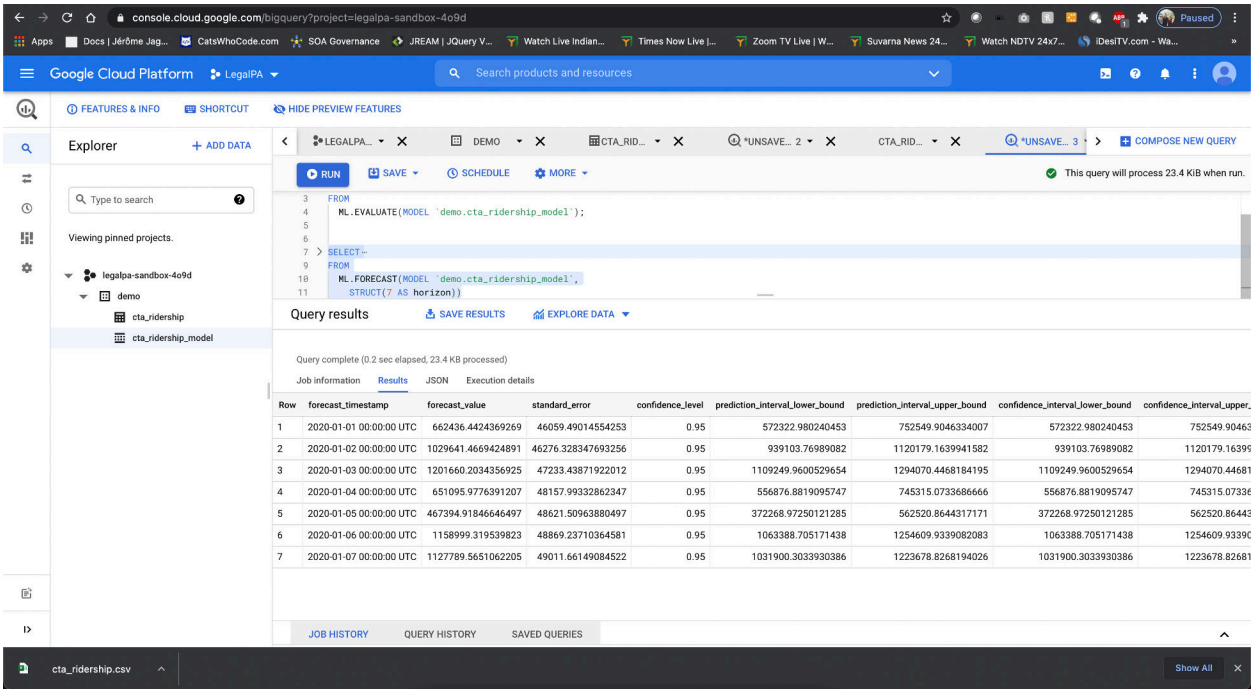

console.cloud.google.com/bigquery?project=legalpa-sandbox-4o9d
Google Cloud Platform
LegalPA
Search products and resources
FEATURES & INFO
SHORTCUT
HIDE PREVIEW FEATURES
Explorer
ADD DATA
Type to search
Viewing pinned projects.
legalpa-sandbox-4o9d
demo
cta_ridership
cta_ridership_model
COMPOSE NEW QUERY
RUN
SAVE
SCHEDULE
MORE
This query will process 108.4 KiB (ML) when run.
Query results
Query complete (2 min 50 sec elapsed, 4.4 MB (ML) processed)
Job information
Results
Execution details
Elapsed time
2 min 50 sec
Slot time consumed
18 min 55.432 sec
Stages
Preprocess 20.170 sec
Train 1 min 23.733 sec
Evaluate 1 min 5.487 sec
Training iterations
Completed: 1
Planned: 20
Loss
Duration (seconds)
JOB HISTORY
QUERY HISTORY
SAVED QUERIES
cta_ridership.csv
Show All


console.cloud.google.com/bigquery?project=legalpa-sandbox-4o9d&d=demo&m=cta_ridership_model&p=legalpa-sandbox-4o9d&page=model
Google Cloud Platform
LegalPA
Search products and resources
FEATURES & INFO
SHORTCUT
HIDE PREVIEW FEATURES
Explorer
ADD DATA
Type to search
Viewing pinned projects.
legalpa-sandbox-4o9d
demo
cta_ridership
cta_ridership_model
COMPOSE NEW QUERY
cta_ridership_model
QUERY MODEL
DELETE MODEL
EXPORT MODEL
DETAILS
TRAINING
EVALUATION
SCHEMA
ARIMA
US
Model Details
EDIT
Model ID legalpa-sandbox-4o9d:demo.cta_ridership_model
Description
Labels
Date created Saturday, March 6, 2021 at 12:07:58 AM GMT-08:00
Model expiration Never
Date modified Saturday, March 6, 2021 at 12:07:58 AM GMT-08:00
Data location US
Model type ARIMA
Training Options
Training options are the optional parameters that were added in the script to create this model.
Actual iterations 1
Auto Arima true
Data Frequency Auto Frequency
Holiday Region US
Auto Arima Max Order 5
JOB HISTORY
QUERY HISTORY
SAVED QUERIES
cta_ridership.csv
Show All

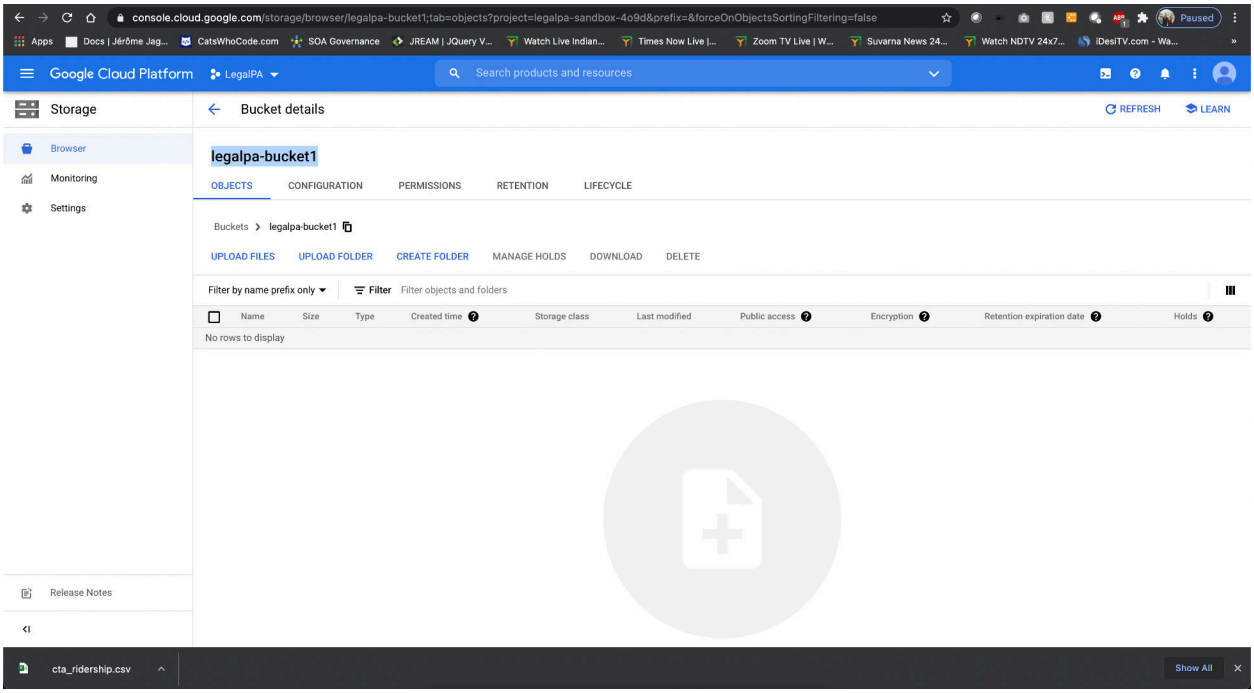
Step 4

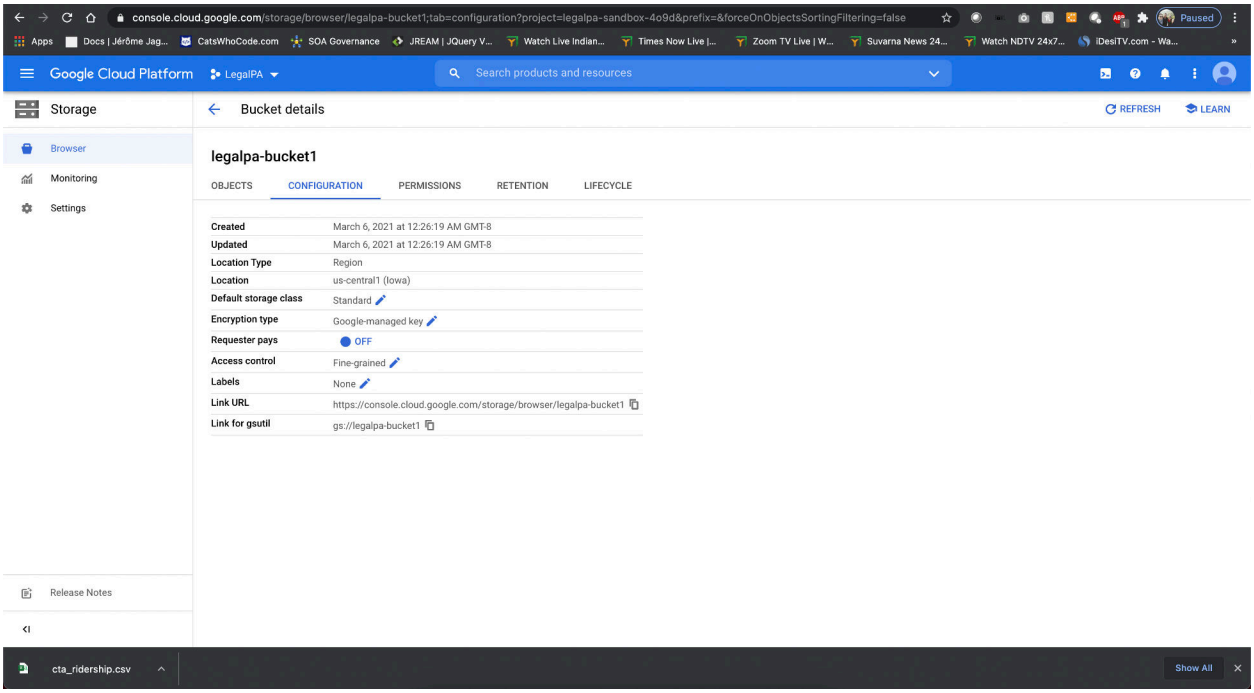

console.cloud.google.com/bigquery?project=legalpa-sandbox-4o9d
Google Cloud Platform
LegalPA
Search products and resources
FEATURES & INFO
SHORTCUT
HIDE PREVIEW FEATURES
Explorer
ADD DATA
Type to search
Viewing pinned projects.
legalpa-sandbox-4o9d
demo
cta_ridership
cta_ridership_model
COMPOSE NEW QUERY
RUN
SAVE
SCHEDULE
MORE
This query will process 0 B when run.
SELECT
*
FROM
ML.EVALUATE(MODEL `demo.cta_ridership_model`)
Query results
SAVE RESULTS
EXPLORE DATA
Query complete (0.3 sec elapsed, 0 B processed)
Job information
Results
JSON
Execution details
Row non_seasonal_p non_seasonal_d non_seasonal_q has_drift log_likelihood AIC variance seasonal_periods
1 1 1 4 true -84343.91298029698 168701.82596059397 2.1214766324672794E9 WEEKLY
YEARLY
2 1 1 4 false -84345.76278035615 168703.5255607123 2.1226282591786644E9 WEEKLY
YEARLY
3 4 1 1 true -84346.86918283005 168707.7383656601 2.1232853081307085E9 WEEKLY
YEARLY
4 1 1 3 true -84347.97278479983 168707.94556959966 2.1239599007139666E9 WEEKLY
YEARLY
Rows per page: 100
1 - 42 of 42
JOB HISTORY
QUERY HISTORY
SAVED QUERIES
cta_ridership.csv
Show All

Step5:

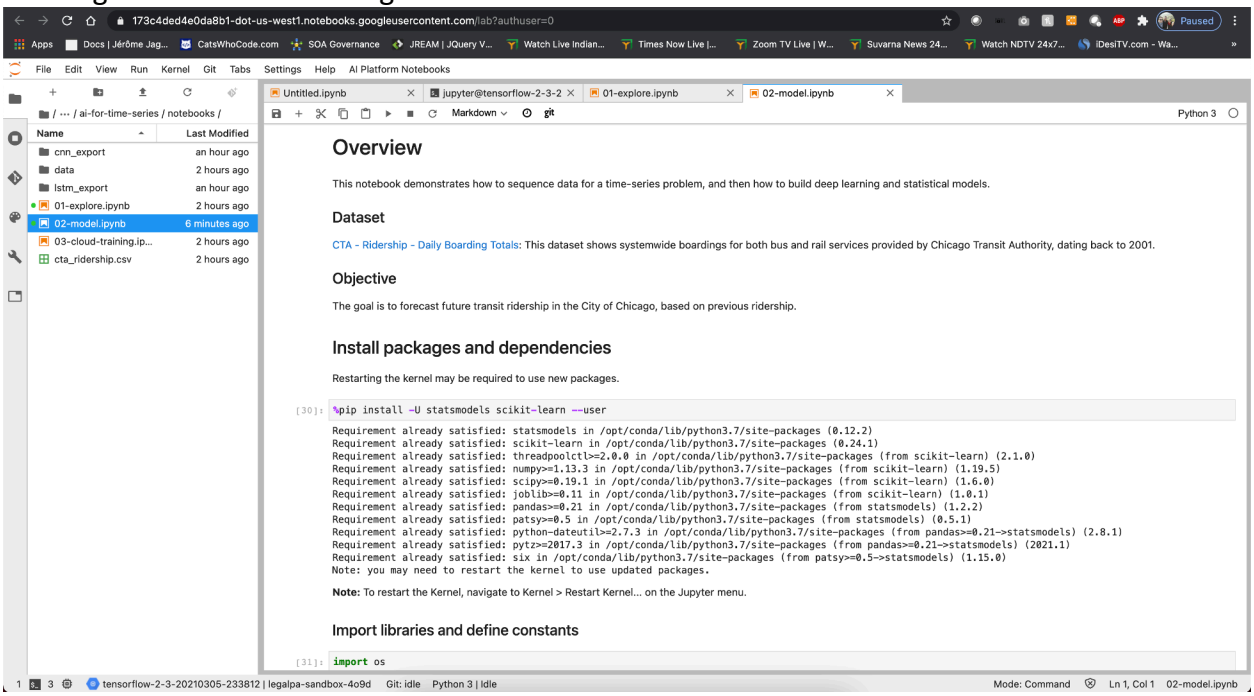


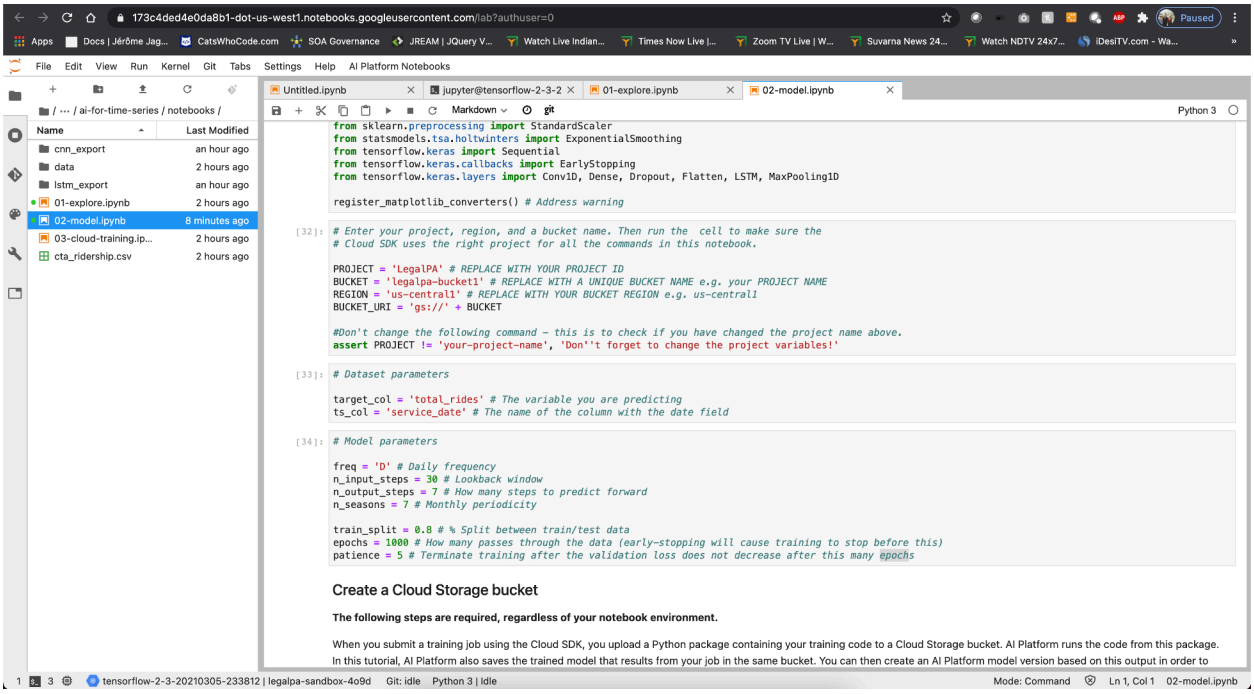
Create a new Bucket

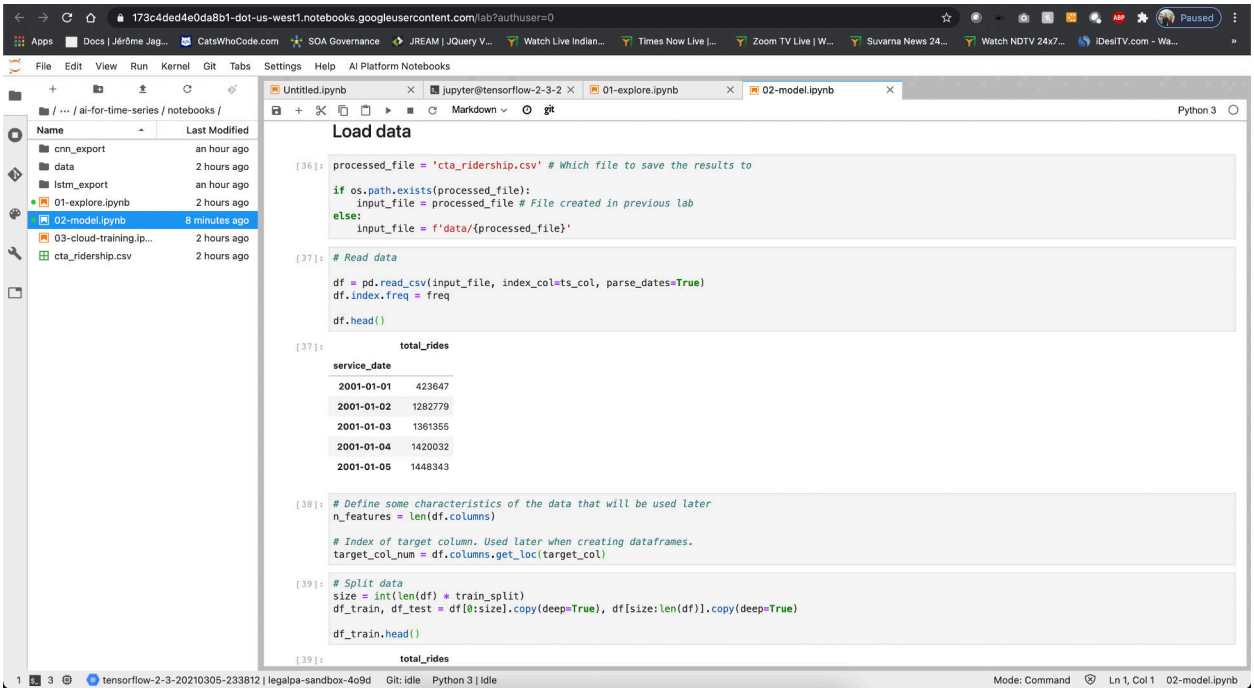




Building a Custom Forecasting Model




173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0
Paused
Apps
Docs | Jérôme Jag...
CatsWhoCode.com
SOA Governance
JREAM | JQuery V...
Watch Live Indian...
Times Now Live |...
Zoom TV Live | W...
Suvarna News 24...
Watch NDTV 24x7...
iDesiTV.com - Wa...
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
/ ··· / ai-for-time-series / notebooks /
Name
Last Modified
cnn_export an hour ago
data 2 hours ago
lstm_export an hour ago
01-explore.ipynb 2 hours ago
02-model.ipynb 8 minutes ago
03-cloud-training.ip... 2 hours ago
cta_ridership.csv 2 hours ago
Untitled.ipynb
jupyter@tensorflow-2-3-2
01-explore.ipynb
02-model.ipynb
Markdown
git
Python 3
from sklearn.preprocessing import StandardScaler
from statsmodels.tsa.holtwinters import ExponentialSmoothing
from tensorflow.keras import Sequential
from tensorflow.keras.callbacks import EarlyStopping
from tensorflow.keras.layers import Conv1D, Dense, Dropout, Flatten, LSTM, MaxPooling1D
register_matplotlib_converters() # Address warning
[32]:
# Enter your project, region, and a bucket name. Then run the cell to make sure the
# Cloud SDK uses the right project for all the commands in this notebook.
PROJECT = 'LegalPA' # REPLACE WITH YOUR PROJECT ID
BUCKET = 'legalpa-bucket1' # REPLACE WITH A UNIQUE BUCKET NAME e.g. your PROJECT NAME
REGION = 'us-central1' # REPLACE WITH YOUR BUCKET REGION e.g. us-central1
BUCKET_URI = 'gs://' + BUCKET
#Don't change the following command - this is to check if you have changed the project name above.
assert PROJECT != 'your-project-name', 'Don''t forget to change the project variables!'
[33]:
# Dataset parameters
target_col = 'total_rides' # The variable you are predicting
ts_col = 'service_date' # The name of the column with the date field
[34]:
# Model parameters
freq = 'D' # Daily frequency
n_input_steps = 30 # Lookback window
n_output_steps = 7 # How many steps to predict forward
n_seasons = 7 # Monthly periodicity
train_split = 0.8 # % Split between train/test data
epochs = 1000 # How many passes through the data (early-stopping will cause training to stop before this)
patience = 5 # Terminate training after the validation loss does not decrease after this many epochs
Create a Cloud Storage bucket
The following steps are required, regardless of your notebook environment.
When you submit a training job using the Cloud SDK, you upload a Python package containing your training code to a Cloud Storage bucket. AI Platform runs the code from this package. In this tutorial, AI Platform also saves the trained model that results from your job in the same bucket. You can then create an AI Platform model version based on this output in order to
1 3 tensorflow-2-3-20210305-233812 | legalpa-sandbox-4o9d Git: idle Python 3 | Idle
Mode: Command Ln 1, Col 1 02-model.ipynb


173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0
Paused
Apps
Docs | Jérôme Jag...
CatsWhoCode.com
SOA Governance
JREAM | JQuery V...
Watch Live Indian...
Times Now Live |...
Zoom TV Live | W...
Suvarna News 24...
Watch NDTV 24x7...
iDesiTV.com - Wa...
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
/ ··· / ai-for-time-series / notebooks /
Name
Last Modified
cnn_export an hour ago
data 2 hours ago
lstm_export an hour ago
01-explore.ipynb 2 hours ago
02-model.ipynb 8 minutes ago
03-cloud-training.ip... 2 hours ago
cta_ridership.csv 2 hours ago
Untitled.ipynb
jupyter@tensorflow-2-3-2
01-explore.ipynb
02-model.ipynb
Markdown
git
Python 3
Load data
[36]:
processed_file = 'cta_ridership.csv' # Which file to save the results to
if os.path.exists(processed_file):
input_file = processed_file # File created in previous lab
else:
input_file = f'data/{processed_file}'
[37]:
# Read data
df = pd.read_csv(input_file, index_col=ts_col, parse_dates=True)
df.index.freq = freq
df.head()
[37]:
total_rides
service_date
2001-01-01 423647
2001-01-02 1282779
2001-01-03 1361355
2001-01-04 1420032
2001-01-05 1448343
[38]:
# Define some characteristics of the data that will be used later
n_features = len(df.columns)
# Index of target column. Used later when creating dataframes.
target_col_num = df.columns.get_loc(target_col)
[39]:
# Split data
size = int(len(df) * train_split)
df_train, df_test = df[0:size].copy(deep=True), df[size:len(df)].copy(deep=True)
df_train.head()
[39]:
total_rides
1 3 tensorflow-2-3-20210305-233812 | legalpa-sandbox-4o9d Git: idle Python 3 | Idle
Mode: Command Ln 1, Col 1 02-model.ipynb

173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0

Apps | Docs | Jérôme Jag... | CatsWhoCode.com | SOA Governance | JREAM | JQuery V... | Watch Live Indian... | Times Now Live |... | Zoom TV Live | W... | Suvarna News 24... | Watch NDTV 24x7... | iDesiTV.com - Wa...

File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks

/ ··· / ai-for-time-series / notebooks /

Name	Last Modified
cnn_export	an hour ago
data	2 hours ago
lstm_export	an hour ago
01-explore.ipynb	2 hours ago
02-model.ipynb	8 minutes ago
03-cloud-training.ip...	2 hours ago
cta_ridership.csv	2 hours ago

Untitled.ipynb | jupyter@tensorflow-2-3-2 | 01-explore.ipynb | 02-model.ipynb

Markdown | git | Python 3

TODO 1: Remove outliers

- Sometimes, you can improve the accuracy of the model by removing outliers.
- In this lab, you'll simply remove some extremely high values.
- You can also apply techniques such as smoothing or resampling the frequency to reduce the impact of outliers.

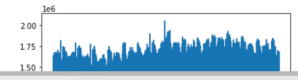
[40]:
```
df_train
```

[40]:

service_date	total_rides
2001-01-01	423647
2001-01-02	1282779
2001-01-03	1361355
2001-01-04	1420032
2001-01-05	1448343
...	...
2016-03-09	1650991
2016-03-10	1668771
2016-03-11	1684576
2016-03-12	1147428
2016-03-13	635332

5551 rows × 1 columns

[41]:
```
# Look at the highest peak. What level could you set a threshold that would clip this off?

_=sns.lineplot(data=df_train[target_col])
```



1 3 tensorflow-2-3-20210305-233812 | legalpa-sandbox-4o9d Git: idle Python 3 | Idle Mode: Command Ln 1, Col 1 02-model.ipynb

173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0

Apps | Docs | Jérôme Jag... | CatsWhoCode.com | SOA Governance | JREAM | JQuery V... | Watch Live Indian... | Times Now Live |... | Zoom TV Live | W... | Suvarna News 24... | Watch NDTV 24x7... | iDesiTV.com - Wa...

File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks

/ ··· / ai-for-time-series / notebooks /

Name	Last Modified
cnn_export	an hour ago
data	2 hours ago
lstm_export	an hour ago
01-explore.ipynb	2 hours ago
02-model.ipynb	9 minutes ago
03-cloud-training.ip...	2 hours ago
cta_ridership.csv	2 hours ago

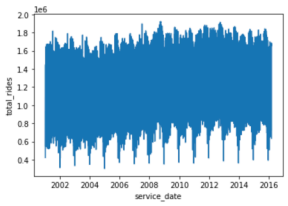
Untitled.ipynb | jupyter@tensorflow-2-3-2 | 01-explore.ipynb | 02-model.ipynb

Markdown | git | Python 3

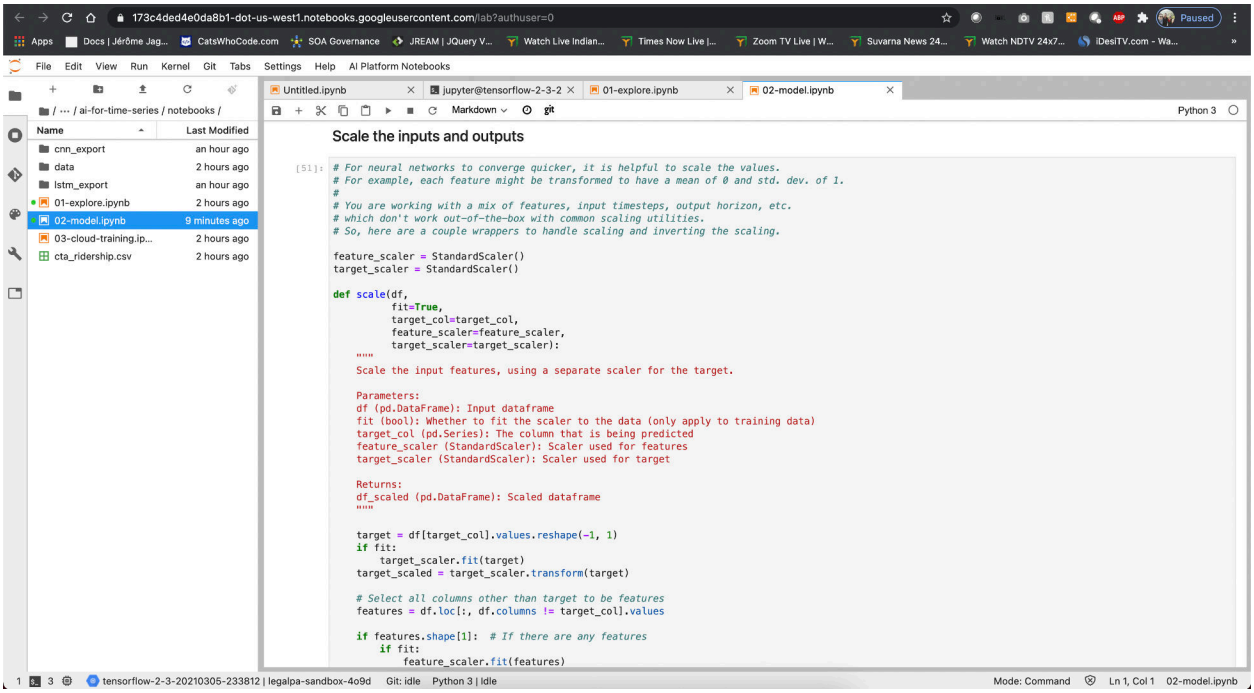
[42]:
```
service_date
2008-07-03    2049519
2016-11-04    1945417
2008-11-04    1938738
2012-10-03    1926454
2008-10-01    1922280
Name: total_rides, dtype: int64
```

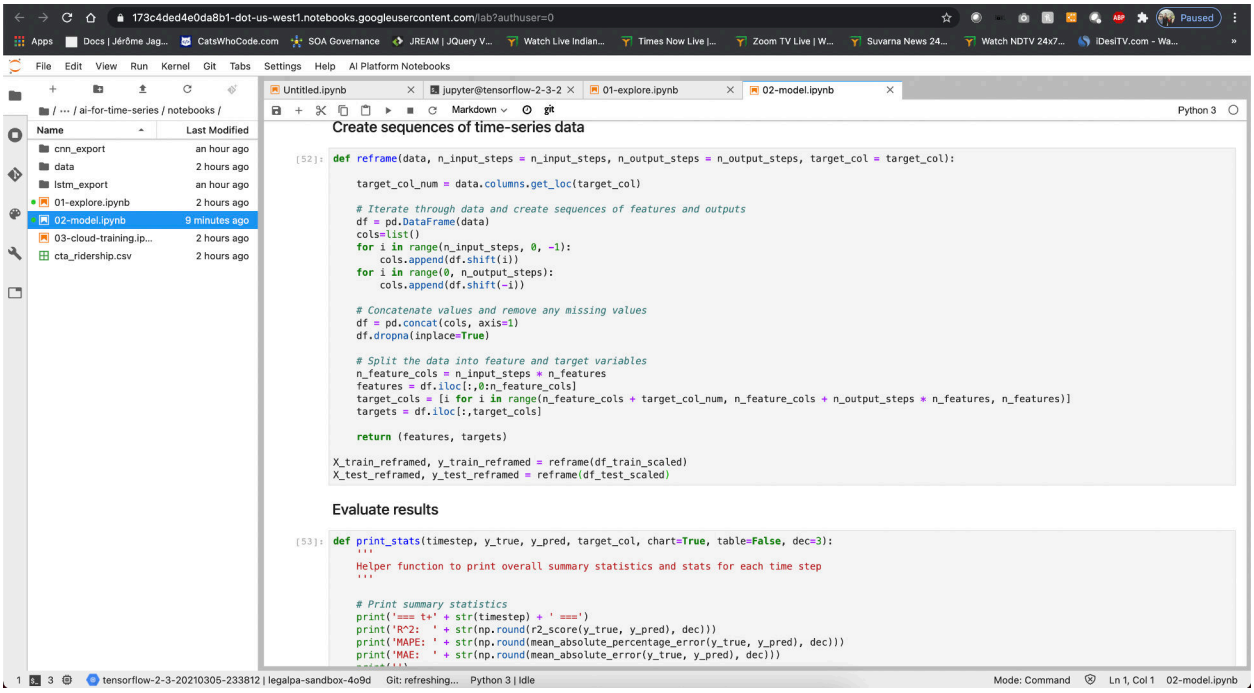
[49]:
```
# TODO: Update the threshold below to remove the outliers

threshold = 3052280 # Set this just below the level you are seeing peaks. It will flag any values above it.
assert threshold != -1, 'Set the threshold to the minimum that will eliminate outlier(s)'

# Set any values above the threshold to NaN (not a number)
df_train.loc[df_train[target_col] > threshold, target_col] = np.nan

# Interpolate the missing values (e.g. [3, NaN, 5] becomes [3, 4, 5])
df_train = df_train.interpolate()
```

[50]:
```
# Review the updated chart to see if outliers still exist
# NOTE: If you set the threshold too low, rerun starting from the

_=sns.lineplot(data=df_train[target_col])
```

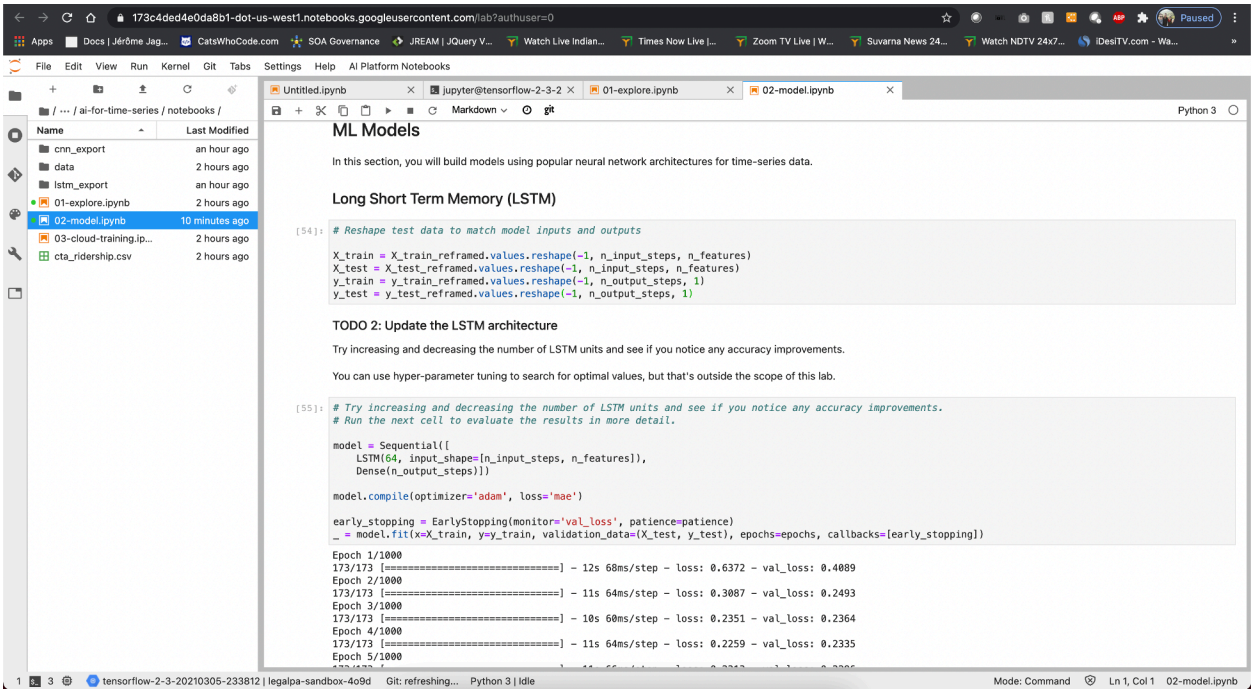


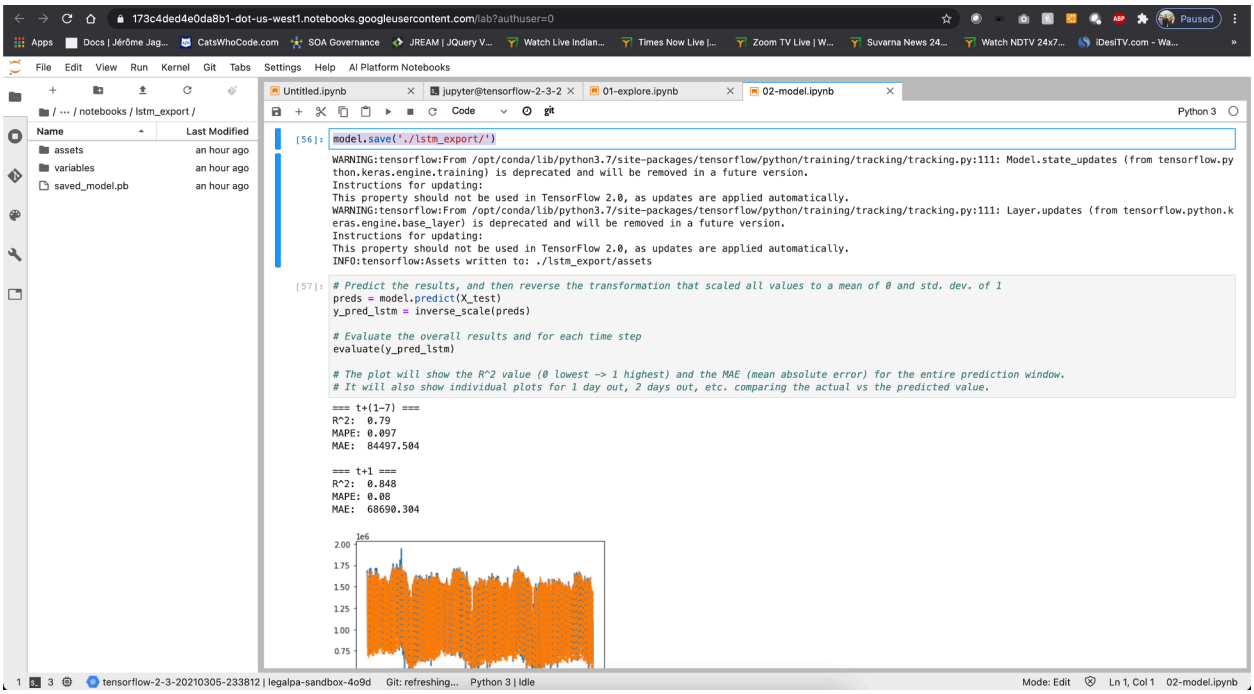
1 3 tensorflow-2-3-20210305-233812 | legalpa-sandbox-4o9d Git: idle Python 3 | Idle Mode: Command Ln 1, Col 1 02-model.ipynb



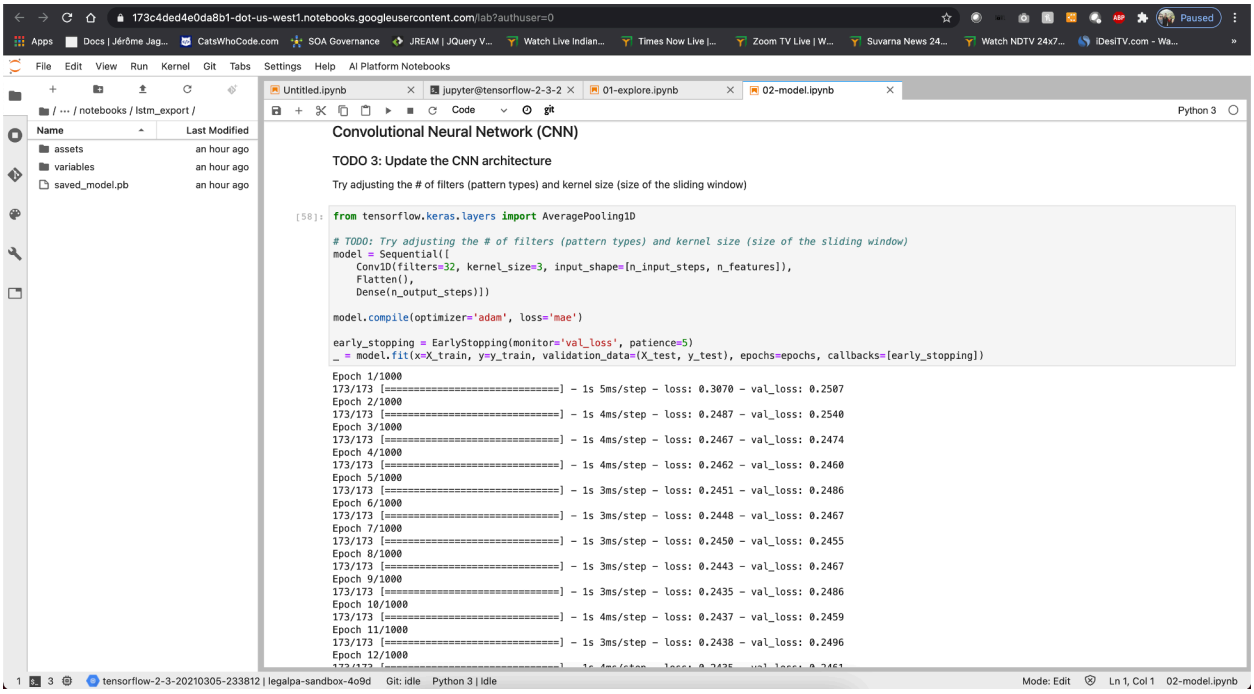


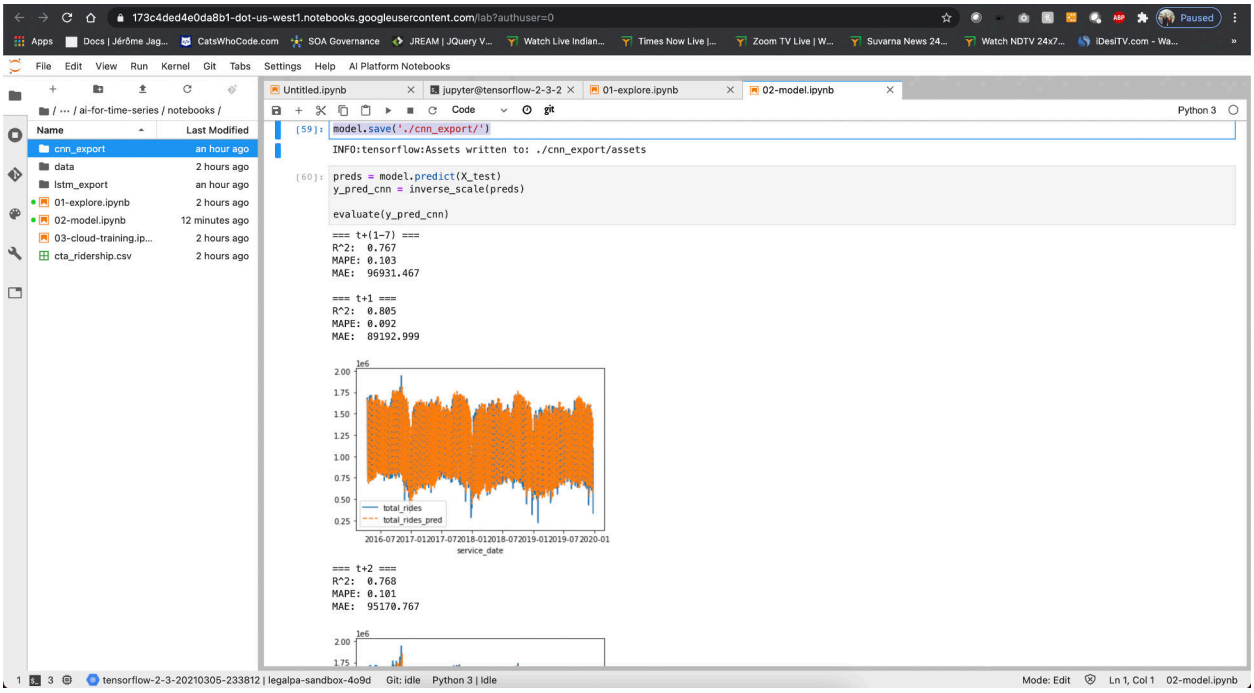
ML: Long Short Term Memory (LSTM) Model



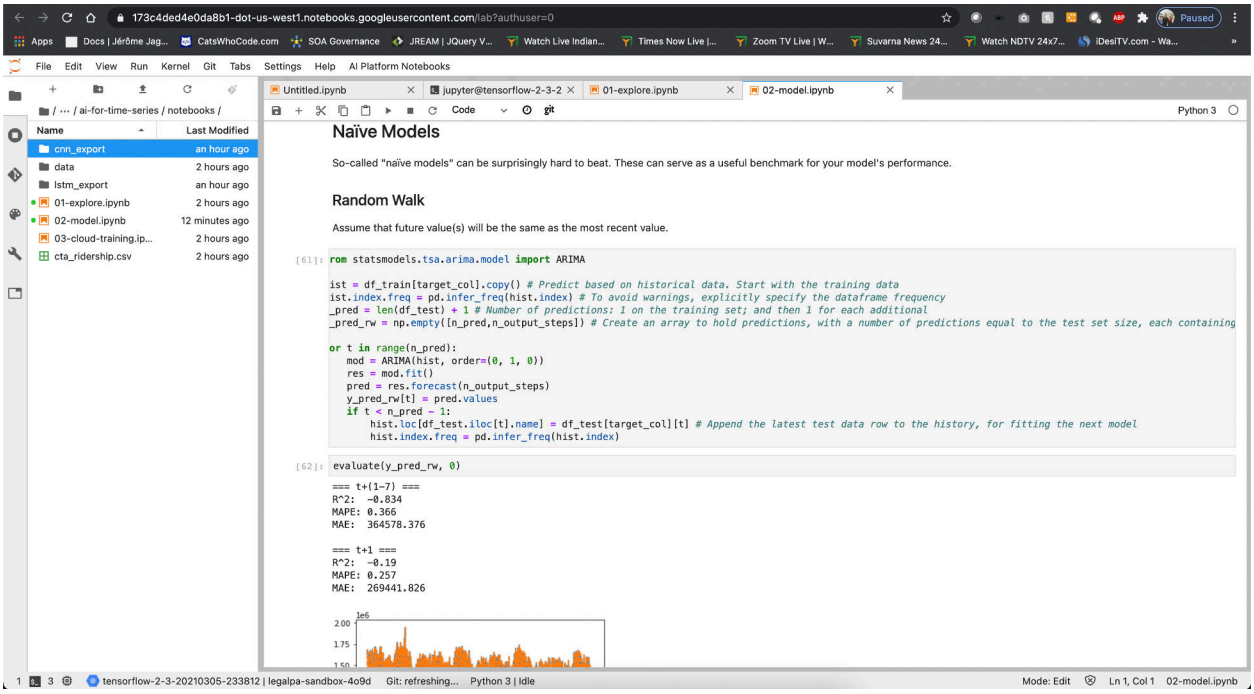


Convolutional Neural Network (CNN) Model

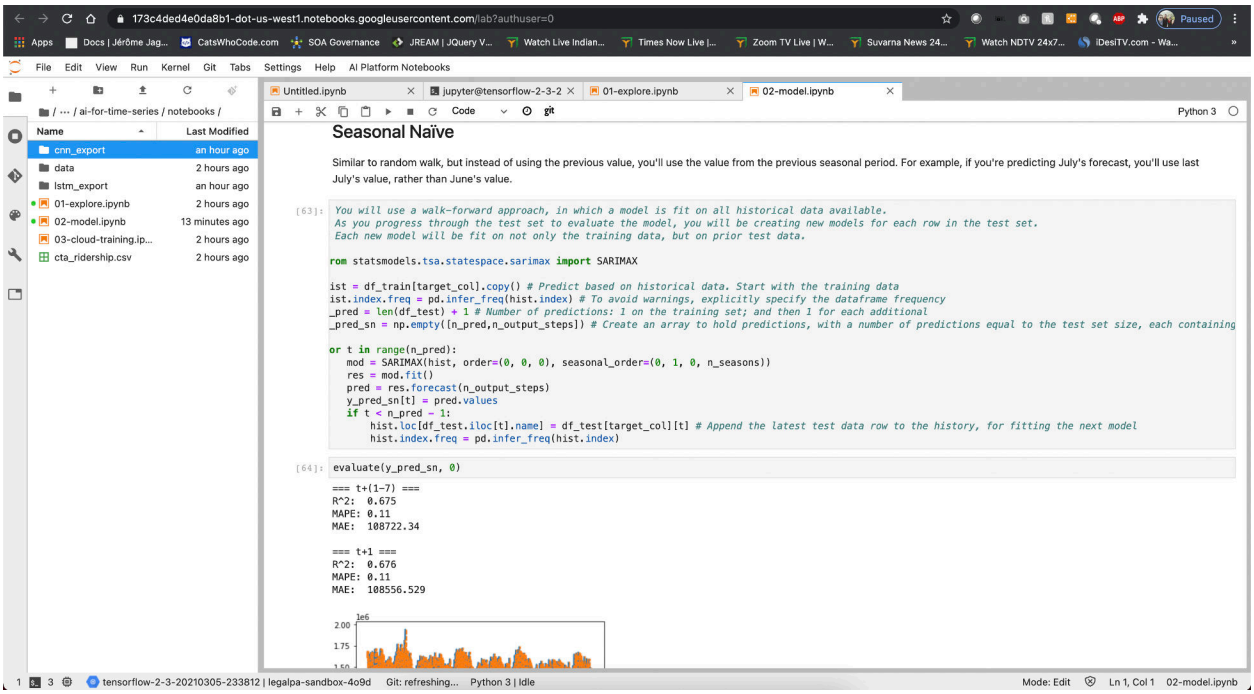

173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0
Paused
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
Untitled.ipynb
jupyter@tensorflow-2-3-2
01-explore.ipynb
02-model.ipynb
Code
git
Python 3
/ ··· / notebooks / lstm_export /
Name
Last Modified
assets an hour ago
variables an hour ago
saved_model.pb an hour ago
Convolutional Neural Network (CNN)
TODO 3: Update the CNN architecture
Try adjusting the # of filters (pattern types) and kernel size (size of the sliding window)
[58]:
from tensorflow.keras.layers import AveragePooling1D
# TODO: Try adjusting the # of filters (pattern types) and kernel size (size of the sliding window)
model = Sequential([
Conv1D(filters=32, kernel_size=3, input_shape=[n_input_steps, n_features]),
Flatten(),
Dense(n_output_steps)])
model.compile(optimizer='adam', loss='mae')
early_stopping = EarlyStopping(monitor='val_loss', patience=5)
_ = model.fit(x=X_train, y=y_train, validation_data=(X_test, y_test), epochs=epochs, callbacks=[early_stopping])
Epoch 1/1000
173/173 [==============================] - 1s 5ms/step - loss: 0.3070 - val_loss: 0.2507
Epoch 2/1000
173/173 [==============================] - 1s 4ms/step - loss: 0.2487 - val_loss: 0.2540
Epoch 3/1000
173/173 [==============================] - 1s 4ms/step - loss: 0.2467 - val_loss: 0.2474
Epoch 4/1000
173/173 [==============================] - 1s 4ms/step - loss: 0.2462 - val_loss: 0.2460
Epoch 5/1000
173/173 [==============================] - 1s 3ms/step - loss: 0.2451 - val_loss: 0.2486
Epoch 6/1000
173/173 [==============================] - 1s 3ms/step - loss: 0.2448 - val_loss: 0.2467
Epoch 7/1000
173/173 [==============================] - 1s 3ms/step - loss: 0.2450 - val_loss: 0.2455
Epoch 8/1000
173/173 [==============================] - 1s 3ms/step - loss: 0.2443 - val_loss: 0.2467
Epoch 9/1000
173/173 [==============================] - 1s 3ms/step - loss: 0.2435 - val_loss: 0.2486
Epoch 10/1000
173/173 [==============================] - 1s 4ms/step - loss: 0.2437 - val_loss: 0.2459
Epoch 11/1000
173/173 [==============================] - 1s 3ms/step - loss: 0.2438 - val_loss: 0.2496
Epoch 12/1000
tensorflow-2-3-20210305-233812 | legalpa-sandbox-4o9d
Git: idle
Python 3 | Idle
Mode: Edit
Ln 1, Col 1
02-model.ipynb


173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0
Paused
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
Untitled.ipynb
jupyter@tensorflow-2-3-2
01-explore.ipynb
02-model.ipynb
Code
git
Python 3
/ ··· / ai-for-time-series / notebooks /
Name
Last Modified
cnn_export an hour ago
data 2 hours ago
lstm_export an hour ago
01-explore.ipynb 2 hours ago
02-model.ipynb 12 minutes ago
03-cloud-training.ip... 2 hours ago
cta_ridership.csv 2 hours ago
[59]:
model.save('./cnn_export/')
INFO:tensorflow:Assets written to: ./cnn_export/assets
[60]:
preds = model.predict(X_test)
y_pred_cnn = inverse_scale(preds)
evaluate(y_pred_cnn)
=== t+(1-7) ===
R^2: 0.767
MAPE: 0.103
MAE: 96931.467
=== t+1 ===
R^2: 0.805
MAPE: 0.092
MAE: 89192.999
1e6
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
total_rides
total_rides_pred
2016-07 2017-01 2017-07 2018-01 2018-07 2019-01 2019-07 2020-01
service_date
=== t+2 ===
R^2: 0.768
MAPE: 0.101
MAE: 95170.767
1e6
2.00
1.75
tensorflow-2-3-20210305-233812 | legalpa-sandbox-4o9d
Git: idle
Python 3 | Idle
Mode: Edit
Ln 1, Col 1
02-model.ipynb

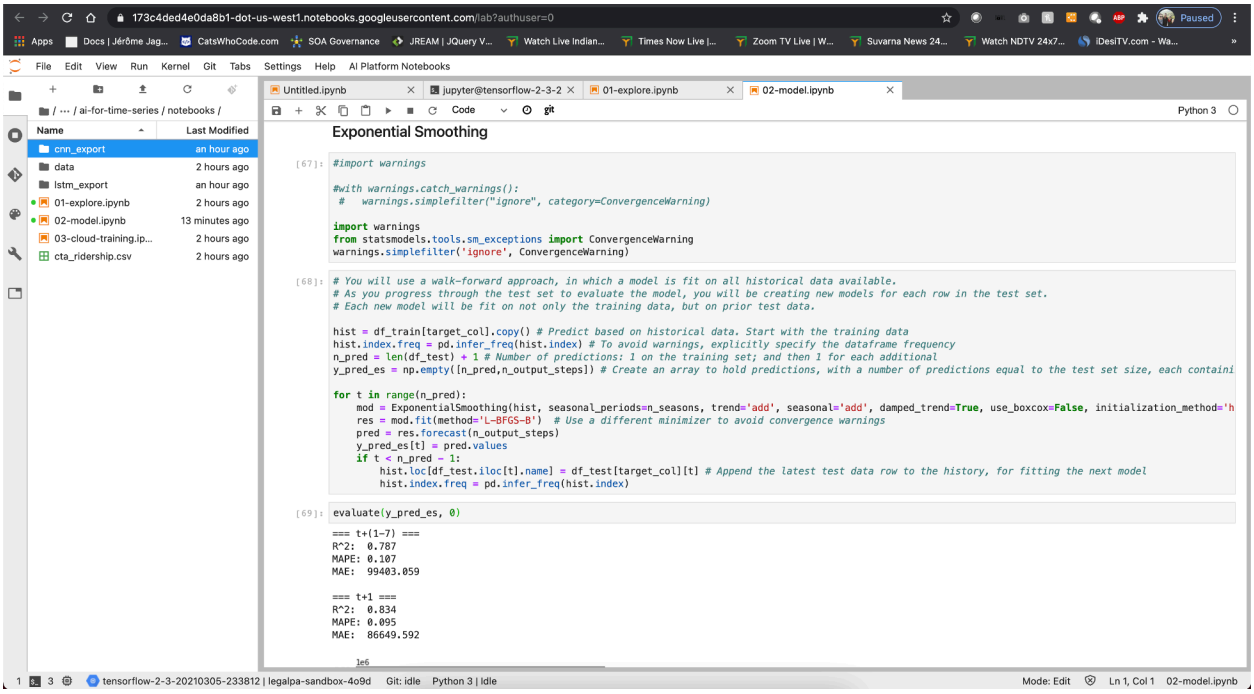
Naïve Model



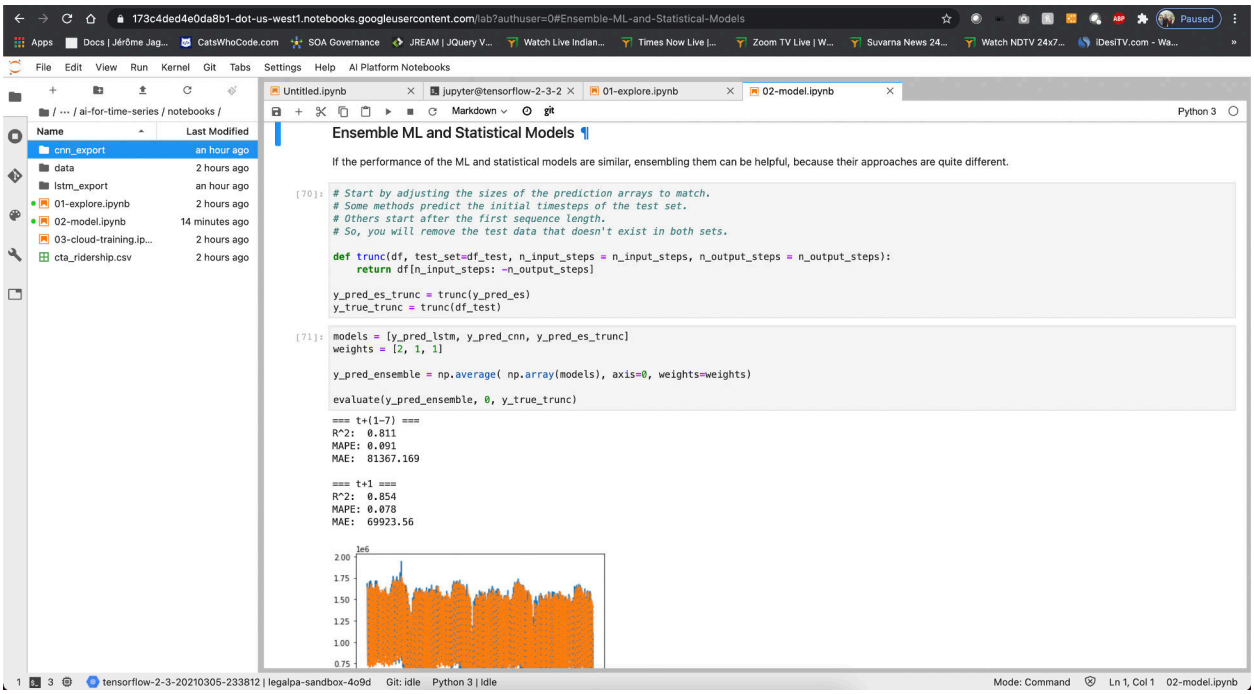
Seasonal Naïve

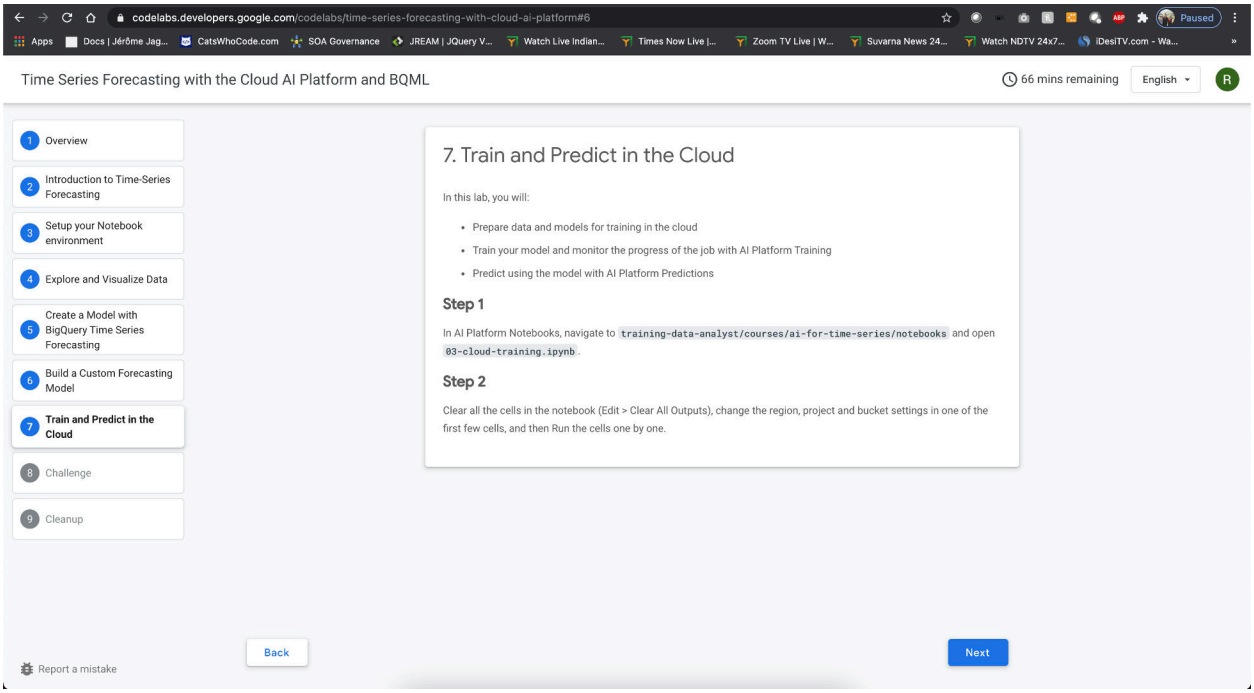


Exponential Smoothing

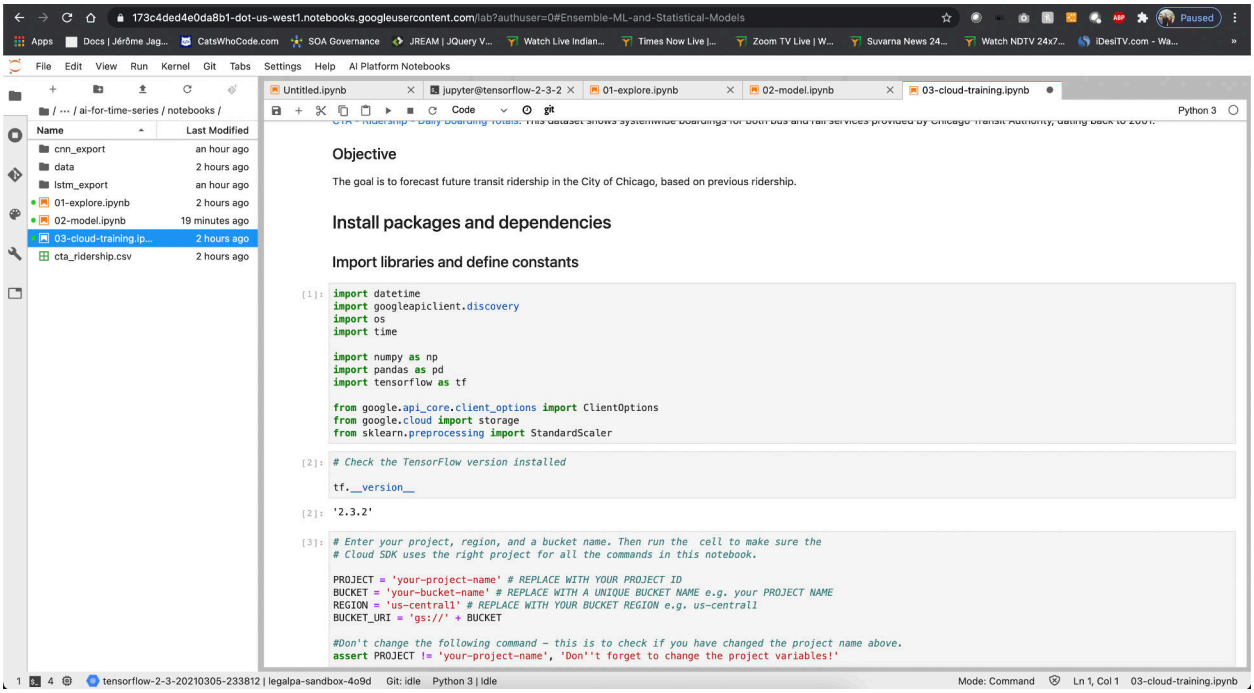


Ensemble ML and Statistical Models





Training and Prediction in Cloud



173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0#Ensemble-ML-and-Statistical-Models

Apps | Docs | Jérôme Jag... | CatsWhoCode.com | SOA Governance | JREAM | JQuery V... | Watch Live Indian... | Times Now Live |... | Zoom TV Live | W... | Suvarna News 24... | Watch NDTV 24x7... | iDesiTV.com - Wa...

File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks

/ ··· / ai-for-time-series / notebooks /

Name	Last Modified
cnn_export	an hour ago
data	2 hours ago
lstm_export	an hour ago
trainer	3 minutes ago
01-explore.ipynb	2 hours ago
02-model.ipynb	23 minutes ago
03-cloud-training.ip...	a minute ago
cta_ridership.csv	2 hours ago

Untitled.ipynb | jupyter@tensorflow-2-3-2 | 01-explore.ipynb | 02-model.ipynb | 03-cloud-training.ipynb

Code | git | Python 3

[7]:
```
# Training parameters

MODEL_NAME = 'cta_ridership'
FRAMEWORK='TENSORFLOW'
RUNTIME_VERSION = '2.3'
PYTHON_VERSION = '3.7'
PREDICTIONS_FILE = 'predictions.json'
```

Create a Cloud Storage bucket

The following steps are required, regardless of your notebook environment.

When you submit a training job using the Cloud SDK, you upload a Python package containing your training code to a Cloud Storage bucket. AI Platform runs the code from this package. In this tutorial, AI Platform also saves the trained model that results from your job in the same bucket. You can then create an AI Platform model version based on this output in order to serve online predictions.

[8]:
```
storage_client = storage.Client()
try:
    bucket = storage_client.get_bucket(BUCKET)
    print('Bucket exists, let''s not recreate it.')
except:
    bucket = storage_client.create_bucket(BUCKET)
    print('Created bucket: ' + BUCKET)
```

```
Bucket exists, lets not recreate it.
```

Load and preview the data

Pre-processing on the original dataset has been done for you and made available on Cloud Storage.

[9]:
```
processed_file = 'cta_ridership.csv' # Which file to save the results to

if os.path.exists(processed_file):
    input_file = processed_file # File created in previous lab
else:
    input_file = f'data/{processed_file}'
```

[10]:
```
df = pd.read_csv(input_file, index_col=ts_col, parse_dates=True)
```

1 4 | tensorflow-2-3-20210305-233812 | legalpa-sandbox-4o9d | Git: idle | Python 3 | Busy | Mode: Command | Ln 1, Col 1 | 03-cloud-training.ipynb

173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0#Ensemble-ML-and-Statistical-Models

Apps | Docs | Jérôme Jag... | CatsWhoCode.com | SOA Governance | JREAM | JQuery V... | Watch Live Indian... | Times Now Live |... | Zoom TV Live | W... | Suvarna News 24... | Watch NDTV 24x7... | iDesiTV.com - Wa...

File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks

/ ··· / ai-for-time-series / notebooks /

Name	Last Modified
cnn_export	an hour ago
data	2 hours ago
lstm_export	an hour ago
trainer	3 minutes ago
01-explore.ipynb	2 hours ago
02-model.ipynb	23 minutes ago
03-cloud-training.ip...	a minute ago
cta_ridership.csv	2 hours ago

Untitled.ipynb | jupyter@tensorflow-2-3-2 | 01-explore.ipynb | 02-model.ipynb | 03-cloud-training.ipynb

Code | git | Python 3

Process data

[12]:
```
# Split data

size = int(len(df) * train_split)
df_train, df_test = df[0:size].copy(deep=True), df[size:len(df)].copy(deep=True)

df_train.head()
```

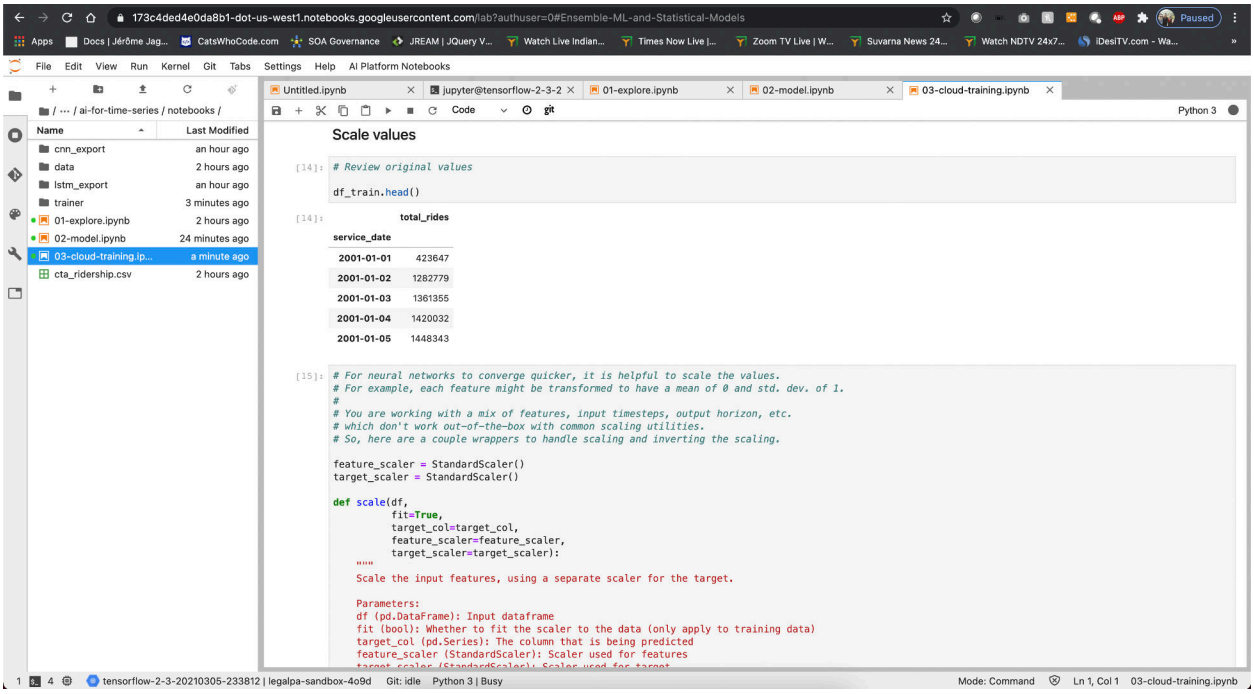
[12]:

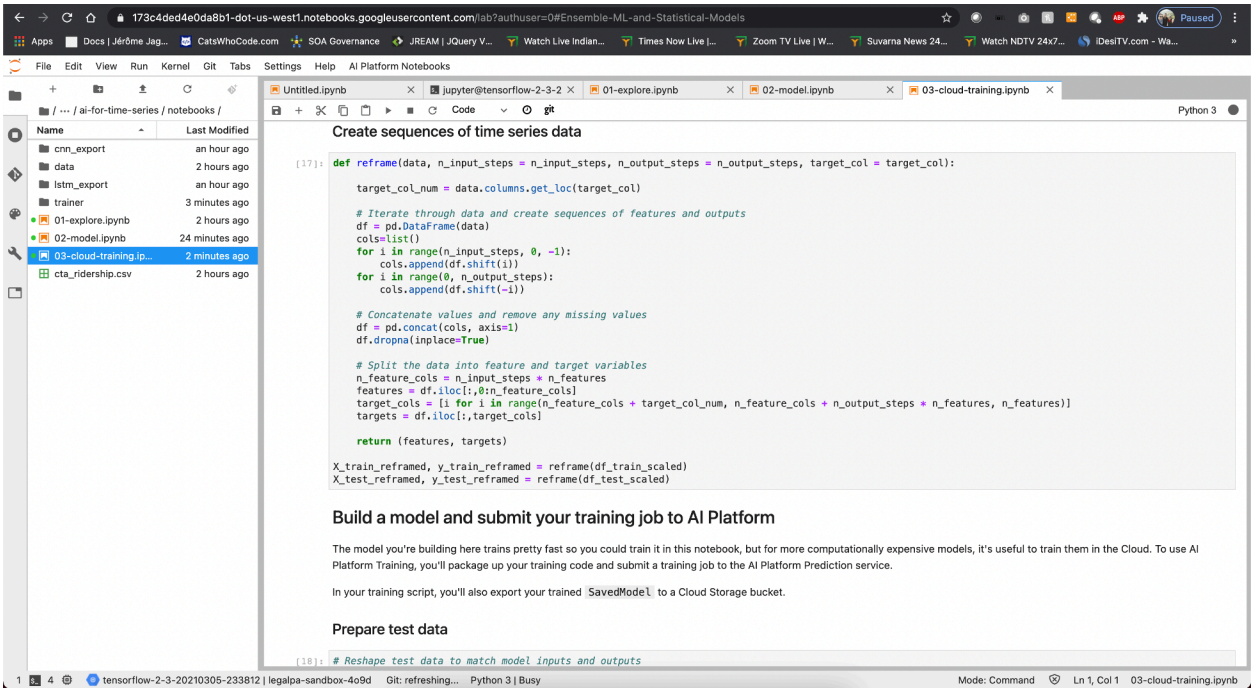
service_date	total_rides
2001-01-01	423647
2001-01-02	1282779
2001-01-03	1361355
2001-01-04	1420032
2001-01-05	1448343

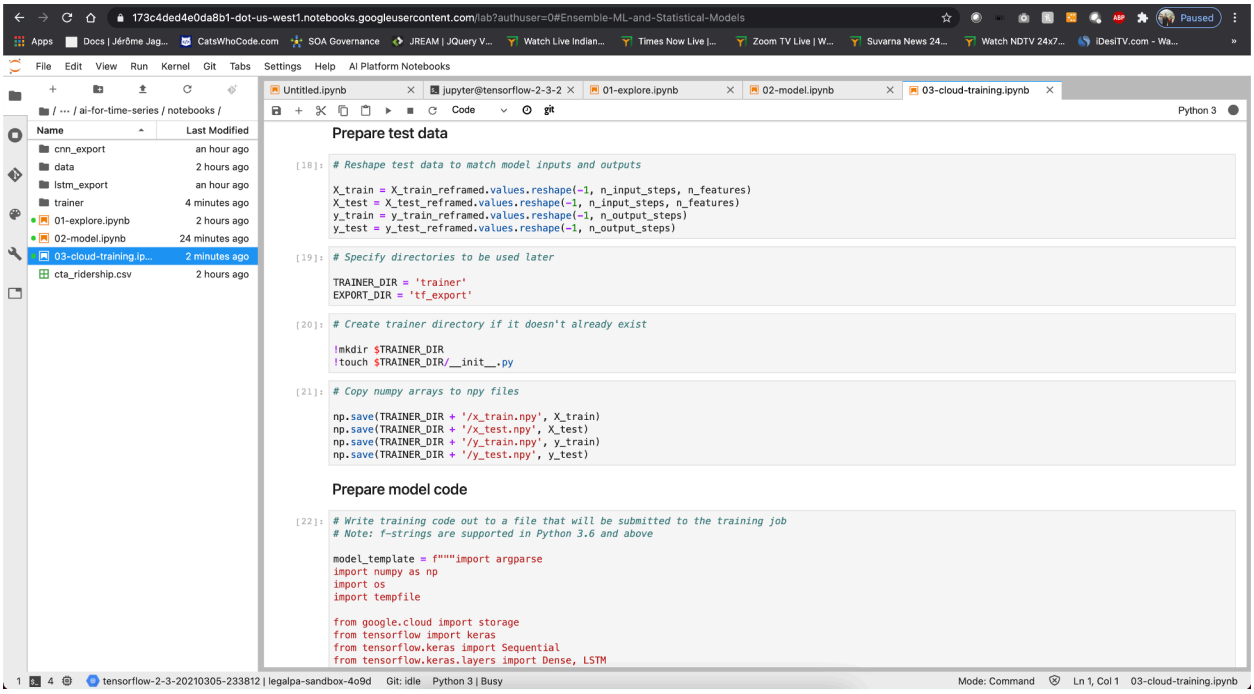
[13]:
```
_ = df_train.plot()
```

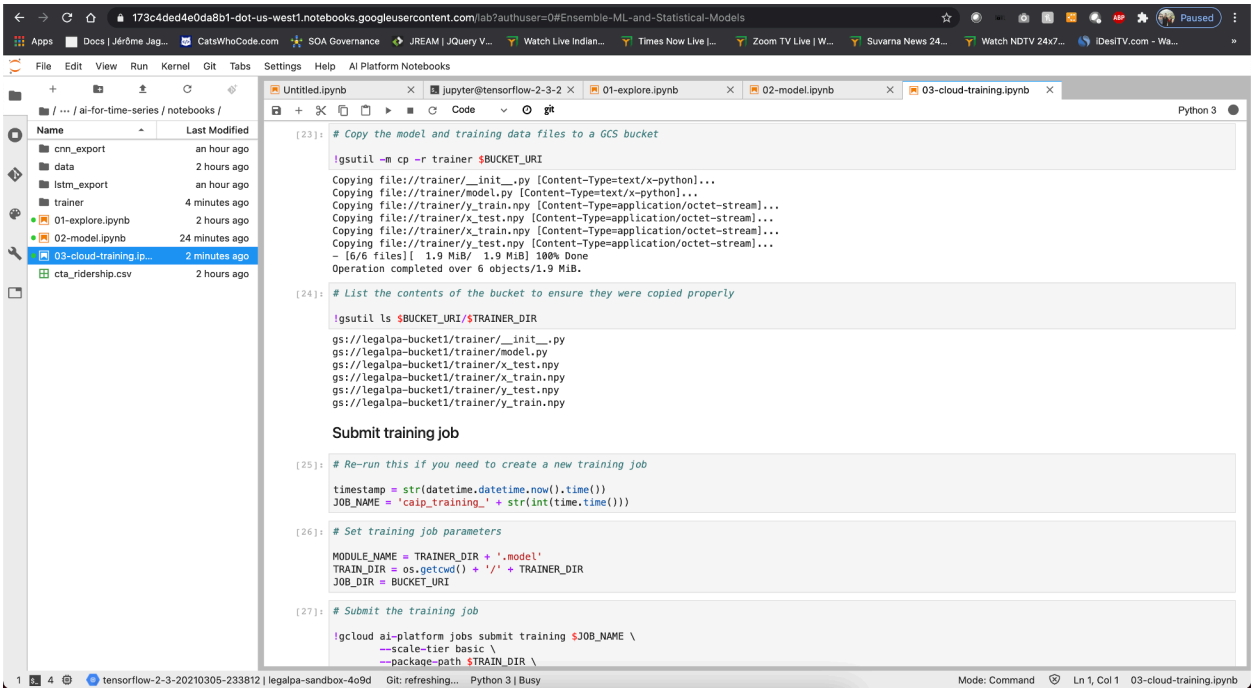
Scale values

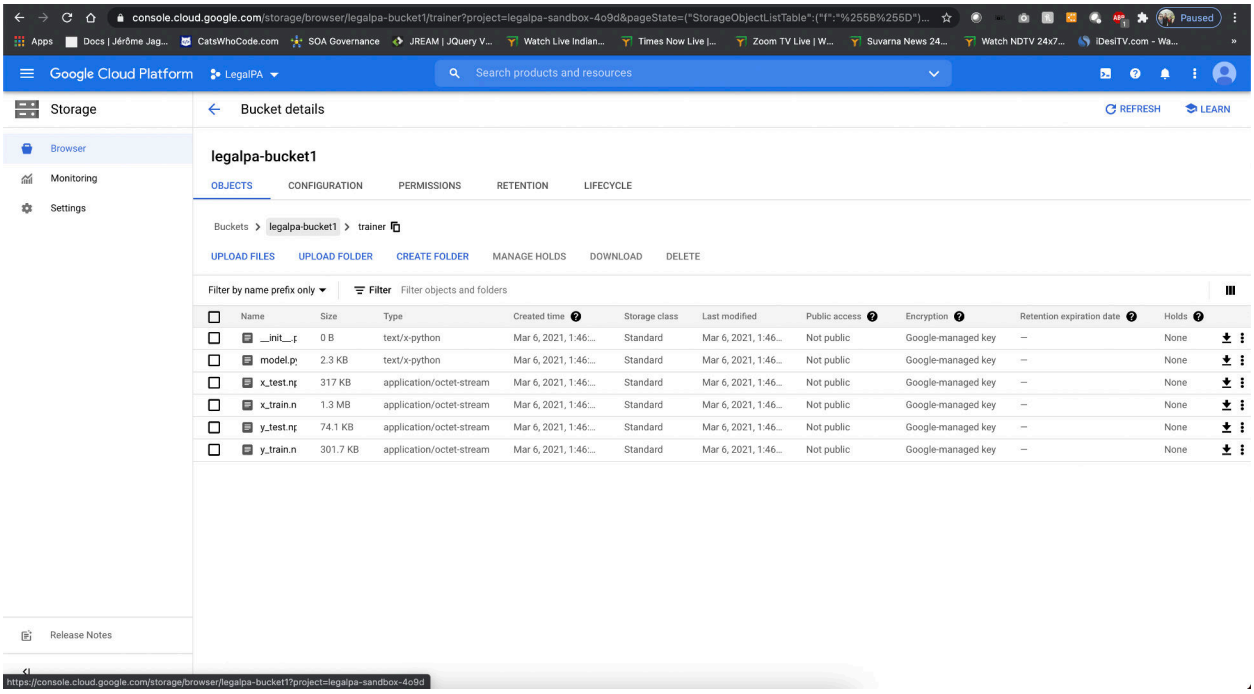
1 4 | tensorflow-2-3-20210305-233812 | legalpa-sandbox-4o9d | Git: refreshing... | Python 3 | Busy | Mode: Command | Ln 1, Col 1 | 03-cloud-training.ipynb

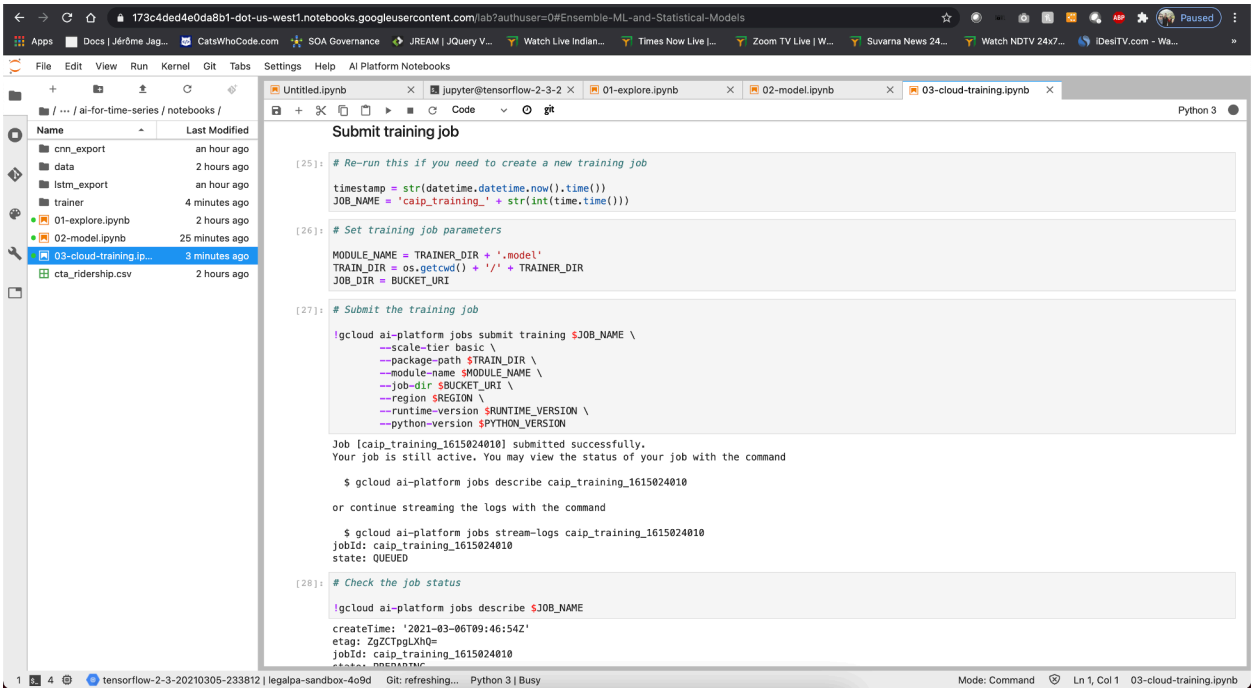

173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0#Ensemble-ML-and-Statistical-Models
Paused
Apps
Docs | Jérôme Jag...
CatsWhoCode.com
SOA Governance
JREAM | JQuery V...
Watch Live Indian...
Times Now Live |...
Zoom TV Live | W...
Suvarna News 24...
Watch NDTV 24x7...
iDesiTV.com - Wa...
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
/ ··· / ai-for-time-series / notebooks /
Name
Last Modified
cnn_export an hour ago
data 2 hours ago
lstm_export an hour ago
trainer 3 minutes ago
01-explore.ipynb 2 hours ago
02-model.ipynb 24 minutes ago
03-cloud-training.ip... a minute ago
cta_ridership.csv 2 hours ago
Untitled.ipynb
jupyter@tensorflow-2-3-2
01-explore.ipynb
02-model.ipynb
03-cloud-training.ipynb
Code
git
Python 3
Scale values
[14]:
# Review original values
df_train.head()
[14]:
total_rides
service_date
2001-01-01 423647
2001-01-02 1282779
2001-01-03 1361355
2001-01-04 1420032
2001-01-05 1448343
[15]:
For neural networks to converge quicker, it is helpful to scale the values.
For example, each feature might be transformed to have a mean of 0 and std. dev. of 1.
#
# You are working with a mix of features, input timesteps, output horizon, etc.
# which don't work out-of-the-box with common scaling utilities.
# So, here are a couple wrappers to handle scaling and inverting the scaling.
feature_scaler = StandardScaler()
target_scaler = StandardScaler()
def scale(df,
fit=True,
target_col=target_col,
feature_scaler=feature_scaler,
target_scaler=target_scaler):
"""
Scale the input features, using a separate scaler for the target.
Parameters:
df (pd.DataFrame): Input dataframe
fit (bool): Whether to fit the scaler to the data (only apply to training data)
target_col (pd.Series): The column that is being predicted
feature_scaler (StandardScaler): Scaler used for features
tensorflow-2-3-20210305-233812 | legalpa-sandbox-4o9d
Git: idle
Python 3 | Busy
Mode: Command
Ln 1, Col 1
03-cloud-training.ipynb

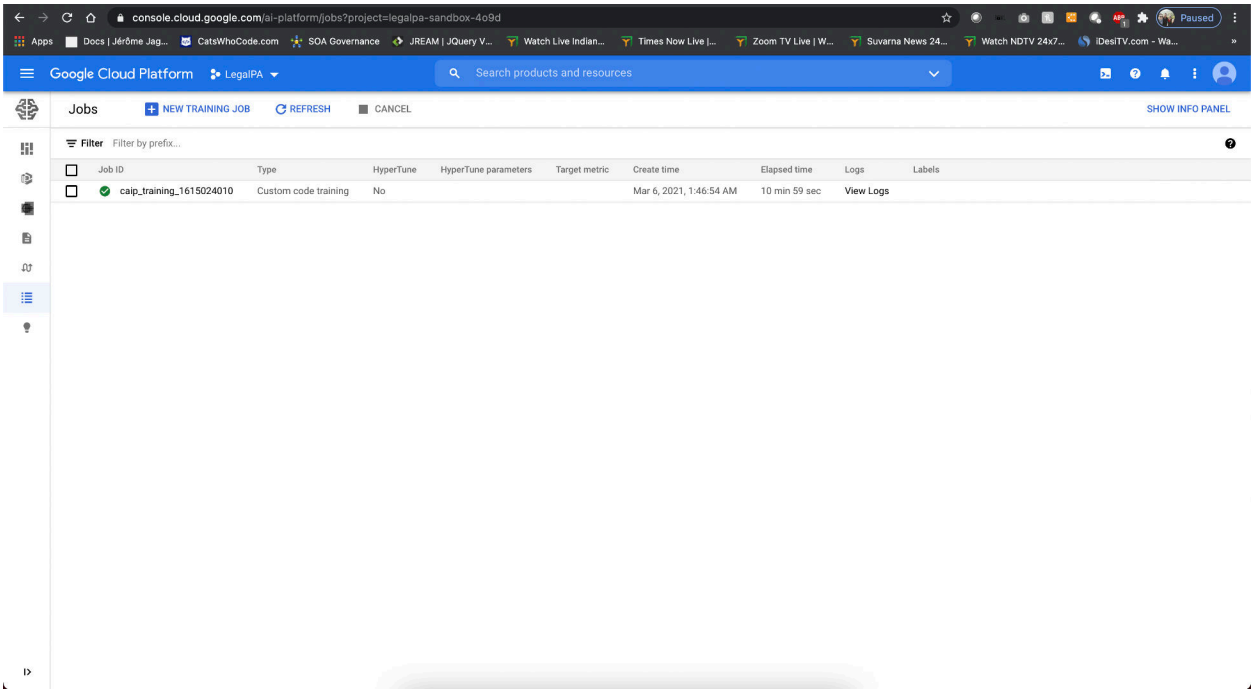

173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0#Ensemble-ML-and-Statistical-Models
Paused
Apps
Docs | Jérôme Jag...
CatsWhoCode.com
SOA Governance
JREAM | JQuery V...
Watch Live Indian...
Times Now Live |...
Zoom TV Live | W...
Suvarna News 24...
Watch NDTV 24x7...
iDesiTV.com - Wa...
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
/ ··· / ai-for-time-series / notebooks /
Name
Last Modified
cnn_export an hour ago
data 2 hours ago
lstm_export an hour ago
trainer 3 minutes ago
01-explore.ipynb 2 hours ago
02-model.ipynb 24 minutes ago
03-cloud-training.ip... 2 minutes ago
cta_ridership.csv 2 hours ago
Untitled.ipynb
jupyter@tensorflow-2-3-2
01-explore.ipynb
02-model.ipynb
03-cloud-training.ipynb
Code
git
Python 3
Create sequences of time series data
[17]:
def reframe(data, n_input_steps = n_input_steps, n_output_steps = n_output_steps, target_col = target_col):
target_col_num = data.columns.get_loc(target_col)
# Iterate through data and create sequences of features and outputs
df = pd.DataFrame(data)
cols=list()
for i in range(n_input_steps, 0, -1):
cols.append(df.shift(i))
for i in range(0, n_output_steps):
cols.append(df.shift(-i))
# Concatenate values and remove any missing values
df = pd.concat(cols, axis=1)
df.dropna(inplace=True)
# Split the data into feature and target variables
n_feature_cols = n_input_steps * n_features
features = df.iloc[:,0:n_feature_cols]
target_cols = [i for i in range(n_feature_cols + target_col_num, n_feature_cols + n_output_steps * n_features, n_features)]
targets = df.iloc[:,target_cols]
return (features, targets)
X_train_reframed, y_train_reframed = reframe(df_train_scaled)
X_test_reframed, y_test_reframed = reframe(df_test_scaled)
Build a model and submit your training job to AI Platform
The model you're building here trains pretty fast so you could train it in this notebook, but for more computationally expensive models, it's useful to train them in the Cloud. To use AI Platform Training, you'll package up your training code and submit a training job to the AI Platform Prediction service.
In your training script, you'll also export your trained SavedModel to a Cloud Storage bucket.
Prepare test data
[18]:
Reshape test data to match model inputs and outputs
tensorflow-2-3-20210305-233812 | legalpa-sandbox-4o9d
Git: refreshing...
Python 3 | Busy
Mode: Command
Ln 1, Col 1
03-cloud-training.ipynb

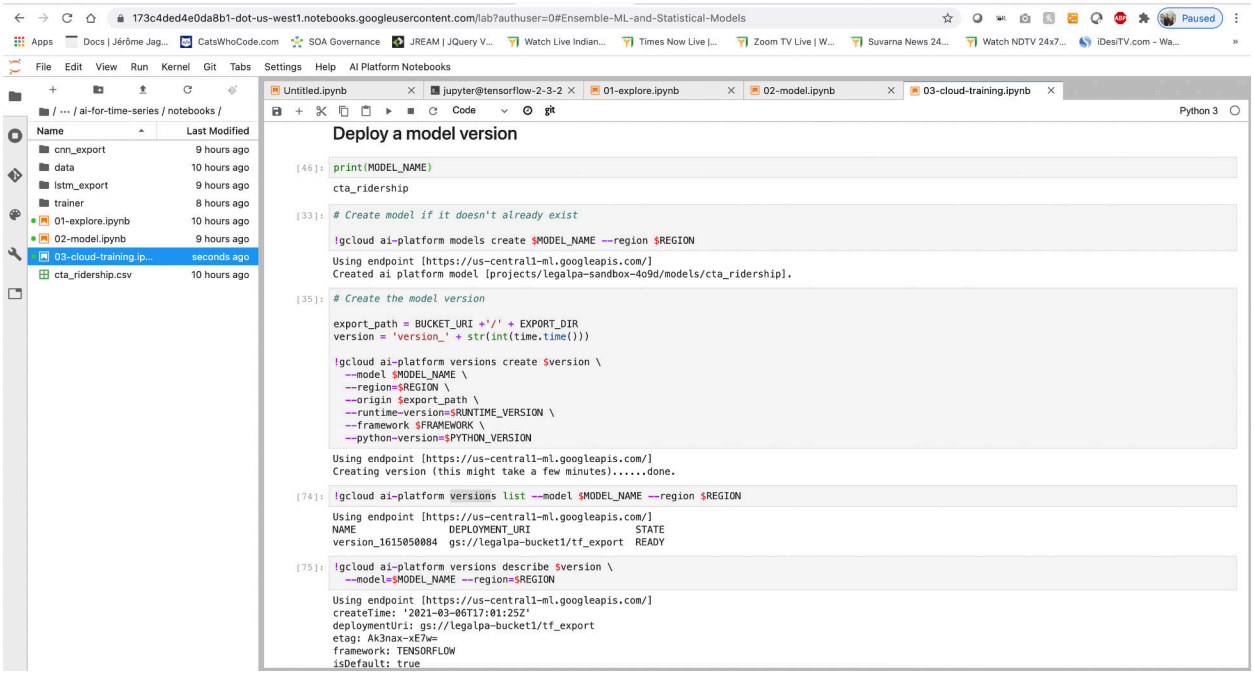

173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0#Ensemble-ML-and-Statistical-Models
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
Untitled.ipynb
jupyter@tensorflow-2-3-2
01-explore.ipynb
02-model.ipynb
03-cloud-training.ipynb
Code
Python 3
/ ··· / ai-for-time-series / notebooks /
Name
Last Modified
cnn_export an hour ago
data 2 hours ago
lstm_export an hour ago
trainer 4 minutes ago
01-explore.ipynb 2 hours ago
02-model.ipynb 24 minutes ago
03-cloud-training.ip... 2 minutes ago
cta_ridership.csv 2 hours ago
Prepare test data
[18]:
# Reshape test data to match model inputs and outputs
X_train = X_train_reframed.values.reshape(-1, n_input_steps, n_features)
X_test = X_test_reframed.values.reshape(-1, n_input_steps, n_features)
y_train = y_train_reframed.values.reshape(-1, n_output_steps)
y_test = y_test_reframed.values.reshape(-1, n_output_steps)
[19]:
# Specify directories to be used later
TRAINER_DIR = 'trainer'
EXPORT_DIR = 'tf_export'
[20]:
Create trainer directory if it doesn't already exist
!mkdir $TRAINER_DIR
!touch $TRAINER_DIR/__init__.py
[21]:
# Copy numpy arrays to npy files
np.save(TRAINER_DIR + '/x_train.npy', X_train)
np.save(TRAINER_DIR + '/x_test.npy', X_test)
np.save(TRAINER_DIR + '/y_train.npy', y_train)
np.save(TRAINER_DIR + '/y_test.npy', y_test)
Prepare model code
[22]:
# Write training code out to a file that will be submitted to the training job
# Note: f-strings are supported in Python 3.6 and above
model_template = f"""import argparse
import numpy as np
import os
import tempfile
from google.cloud import storage
from tensorflow import keras
from tensorflow.keras import Sequential
from tensorflow.keras.layers import Dense, LSTM
tensorflow-2-3-20210305-233812 | legalpa-sandbox-4o9d
Git: idle
Python 3 | Busy
Mode: Command
Ln 1, Col 1
03-cloud-training.ipynb


173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0#Ensemble-ML-and-Statistical-Models
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
Untitled.ipynb
jupyter@tensorflow-2-3-2
01-explore.ipynb
02-model.ipynb
03-cloud-training.ipynb
Code
Python 3
/ ··· / ai-for-time-series / notebooks /
Name
Last Modified
cnn_export an hour ago
data 2 hours ago
lstm_export an hour ago
trainer 4 minutes ago
01-explore.ipynb 2 hours ago
02-model.ipynb 24 minutes ago
03-cloud-training.ip... 2 minutes ago
cta_ridership.csv 2 hours ago
[23]:
Copy the model and training data files to a GCS bucket
!gsutil -m cp -r trainer $BUCKET_URI
Copying file://trainer/__init__.py [Content-Type=text/x-python]...
Copying file://trainer/model.py [Content-Type=text/x-python]...
Copying file://trainer/y_train.npy [Content-Type=application/octet-stream]...
Copying file://trainer/x_test.npy [Content-Type=application/octet-stream]...
Copying file://trainer/x_train.npy [Content-Type=application/octet-stream]...
Copying file://trainer/y_test.npy [Content-Type=application/octet-stream]...
- [6/6 files][1.9 MiB/ 1.9 MiB] 100% Done
Operation completed over 6 objects/1.9 MiB.
[24]:
List the contents of the bucket to ensure they were copied properly
!gsutil ls $BUCKET_URI/$TRAINER_DIR
gs://legalpa-bucket1/trainer/__init__.py
gs://legalpa-bucket1/trainer/model.py
gs://legalpa-bucket1/trainer/x_test.npy
gs://legalpa-bucket1/trainer/x_train.npy
gs://legalpa-bucket1/trainer/y_test.npy
gs://legalpa-bucket1/trainer/y_train.npy
Submit training job
[25]:
# Re-run this if you need to create a new training job
timestamp = str(datetime.datetime.now().time())
JOB_NAME = 'caip_training_' + str(int(time.time()))
[26]:
# Set training job parameters
MODULE_NAME = TRAINER_DIR + '.model'
TRAIN_DIR = os.getcwd() + '/' + TRAINER_DIR
JOB_DIR = BUCKET_URI
[27]:
Submit the training job
!gcloud ai-platform jobs submit training $JOB_NAME \
--scale-tier basic \
--package-path $TRAIN_DIR \
tensorflow-2-3-20210305-233812 | legalpa-sandbox-4o9d
Git: refreshing...
Python 3 | Busy
Mode: Command
Ln 1, Col 1
03-cloud-training.ipynb

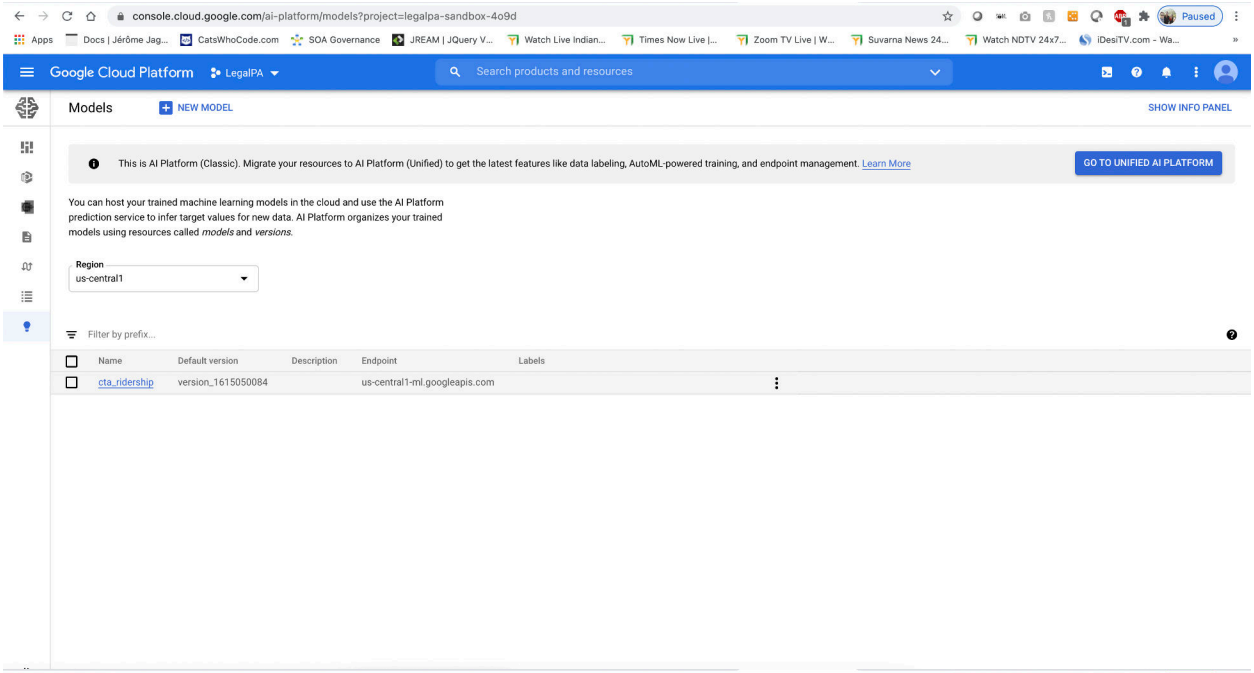

console.cloud.google.com/storage/browser/legalpa-bucket1/trainer?project=legalpa-sandbox-4o9d&pageState=("StorageObjectListTable":("f":"%255B%255D")...
Google Cloud Platform
LegalPA
Search products and resources
Storage
Browser
Monitoring
Settings
Release Notes
Bucket details
REFRESH
LEARN
legalpa-bucket1
OBJECTS
CONFIGURATION
PERMISSIONS
RETENTION
LIFECYCLE
Buckets > legalpa-bucket1 > trainer
UPLOAD FILES
UPLOAD FOLDER
CREATE FOLDER
MANAGE HOLDS
DOWNLOAD
DELETE
Filter by name prefix only
Filter
Filter objects and folders
Name
Size
Type
Created time
Storage class
Last modified
Public access
Encryption
Retention expiration date
Holds
__init__.p
0 B
text/x-python
Mar 6, 2021, 1:46:...
Standard
Mar 6, 2021, 1:46...
Not public
Google-managed key
None
model.p
2.3 KB
text/x-python
x_test.np
317 KB
application/octet-stream
x_train.n
1.3 MB
y_test.np
74.1 KB
y_train.n
301.7 KB
https://console.cloud.google.com/storage/browser/legalpa-bucket1?project=legalpa-sandbox-4o9d

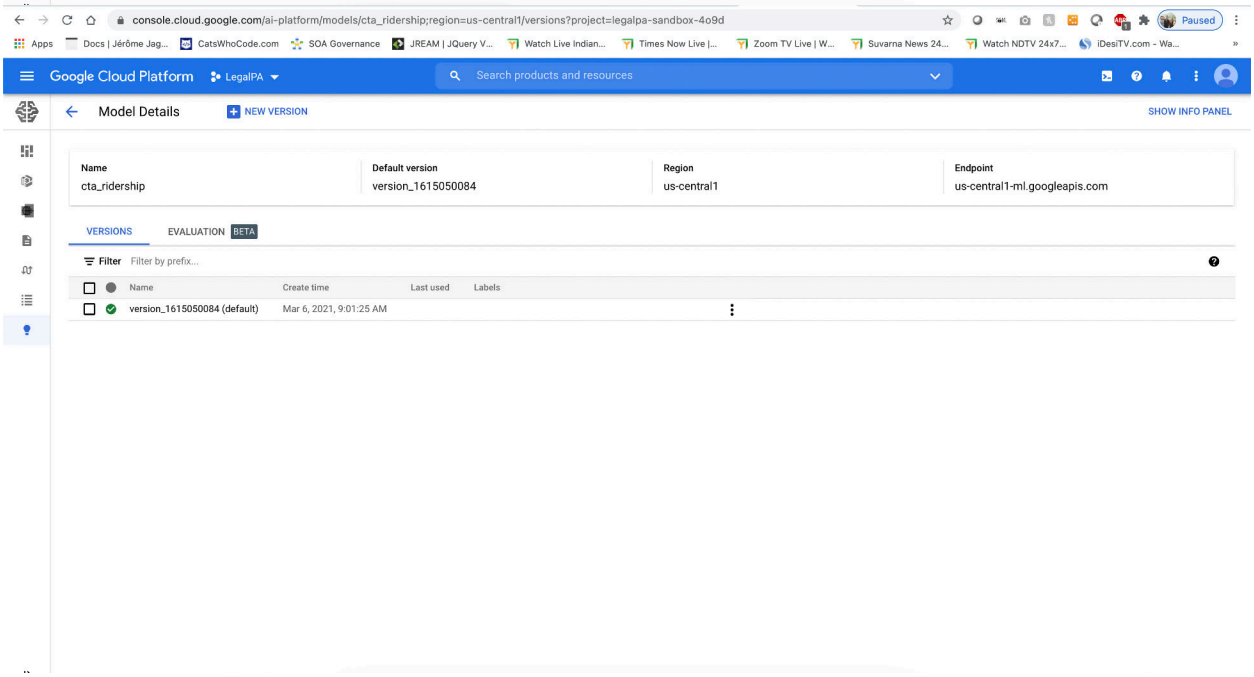

173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0#Ensemble-ML-and-Statistical-Models
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
/ ... / ai-for-time-series / notebooks /
Name
Last Modified
cnn_export an hour ago
data 2 hours ago
lstm_export an hour ago
trainer 4 minutes ago
01-explore.ipynb 2 hours ago
02-model.ipynb 25 minutes ago
03-cloud-training.ip... 3 minutes ago
cta_ridership.csv 2 hours ago
Untitled.ipynb
jupyter@tensorflow-2-3-2
01-explore.ipynb
02-model.ipynb
03-cloud-training.ipynb
Code
git
Python 3
Submit training job
[25]:
# Re-run this if you need to create a new training job
timestamp = str(datetime.datetime.now().time())
JOB_NAME = 'caip_training_' + str(int(time.time()))
[26]:
# Set training job parameters
MODULE_NAME = TRAINER_DIR + '.model'
TRAIN_DIR = os.getcwd() + '/' + TRAINER_DIR
JOB_DIR = BUCKET_URI
[27]:
Submit the training job
!gcloud ai-platform jobs submit training $JOB_NAME \
--scale-tier basic \
--package-path $TRAIN_DIR \
--module-name $MODULE_NAME \
--job-dir $BUCKET_URI \
--region $REGION \
--runtime-version $RUNTIME_VERSION \
--python-version $PYTHON_VERSION
Job [caip_training_1615024010] submitted successfully.
Your job is still active. You may view the status of your job with the command
$ gcloud ai-platform jobs describe caip_training_1615024010
or continue streaming the logs with the command
$ gcloud ai-platform jobs stream-logs caip_training_1615024010
jobId: caip_training_1615024010
state: QUEUED
[28]:
Check the job status
!gcloud ai-platform jobs describe $JOB_NAME
createTime: '2021-03-06T09:46:54Z'
etag: ZgZCTpgLXhQ=
jobId: caip_training_1615024010
tensorflow-2-3-20210305-233812 | legalpa-sandbox-4o9d
Git: refreshing...
Python 3 | Busy
Mode: Command
Ln 1, Col 1
03-cloud-training.ipynb

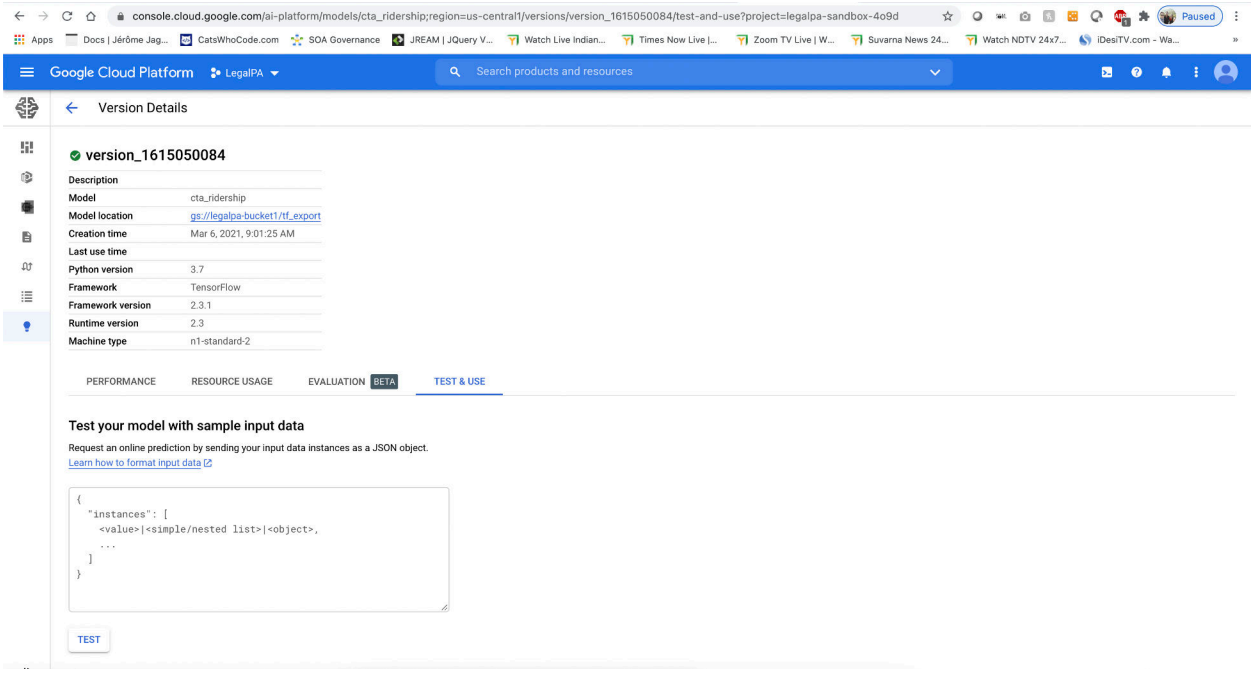




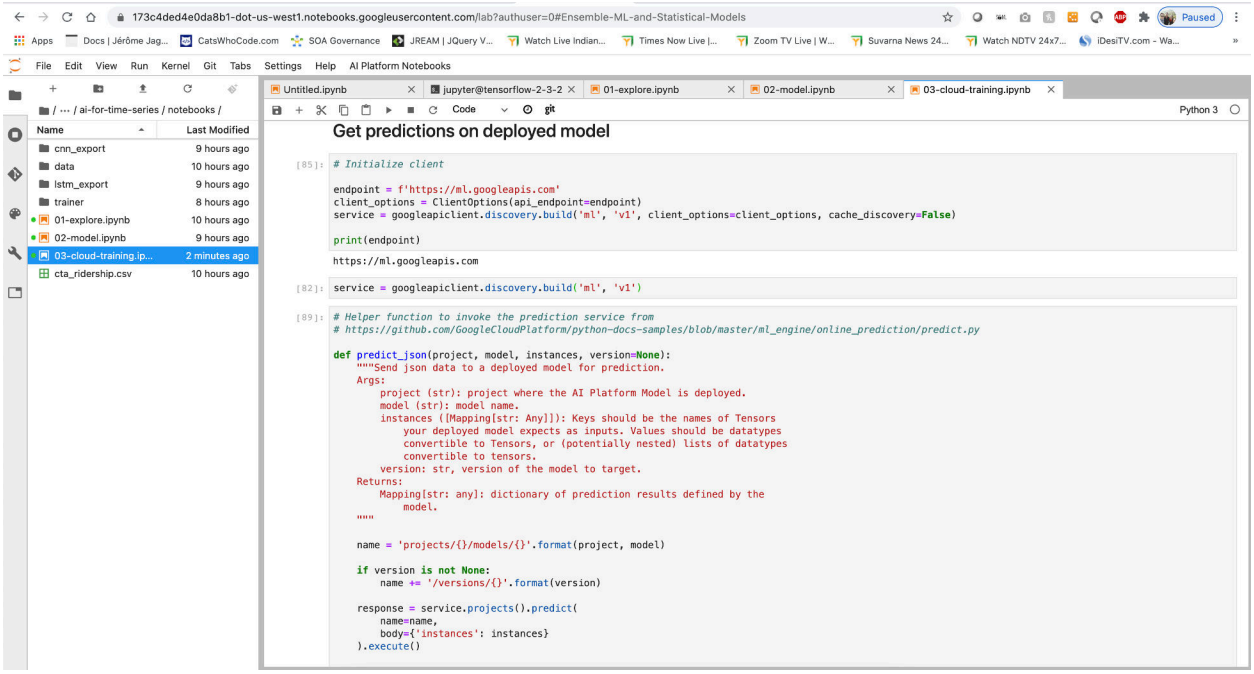
Generated Model:

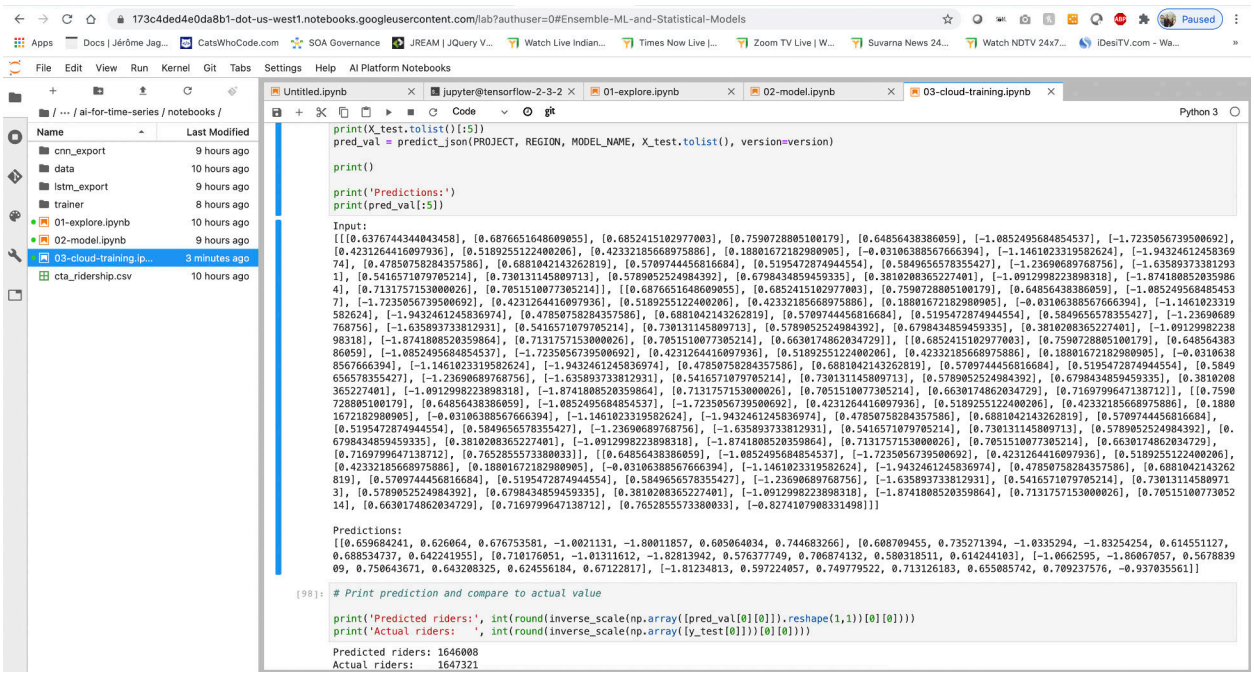

console.cloud.google.com/ai-platform/models?project=legalpa-sandbox-4o9d
Google Cloud Platform
LegalPA
Search products and resources
Models
NEW MODEL
SHOW INFO PANEL
This is AI Platform (Classic). Migrate your resources to AI Platform (Unified) to get the latest features like data labeling, AutoML-powered training, and endpoint management. Learn More
GO TO UNIFIED AI PLATFORM
You can host your trained machine learning models in the cloud and use the AI Platform prediction service to infer target values for new data. AI Platform organizes your trained models using resources called models and versions.
Region
us-central1
Filter by prefix...
Name
Default version
Description
Endpoint
Labels
cta_ridership
version_1615050084
us-central1-ml.googleapis.com

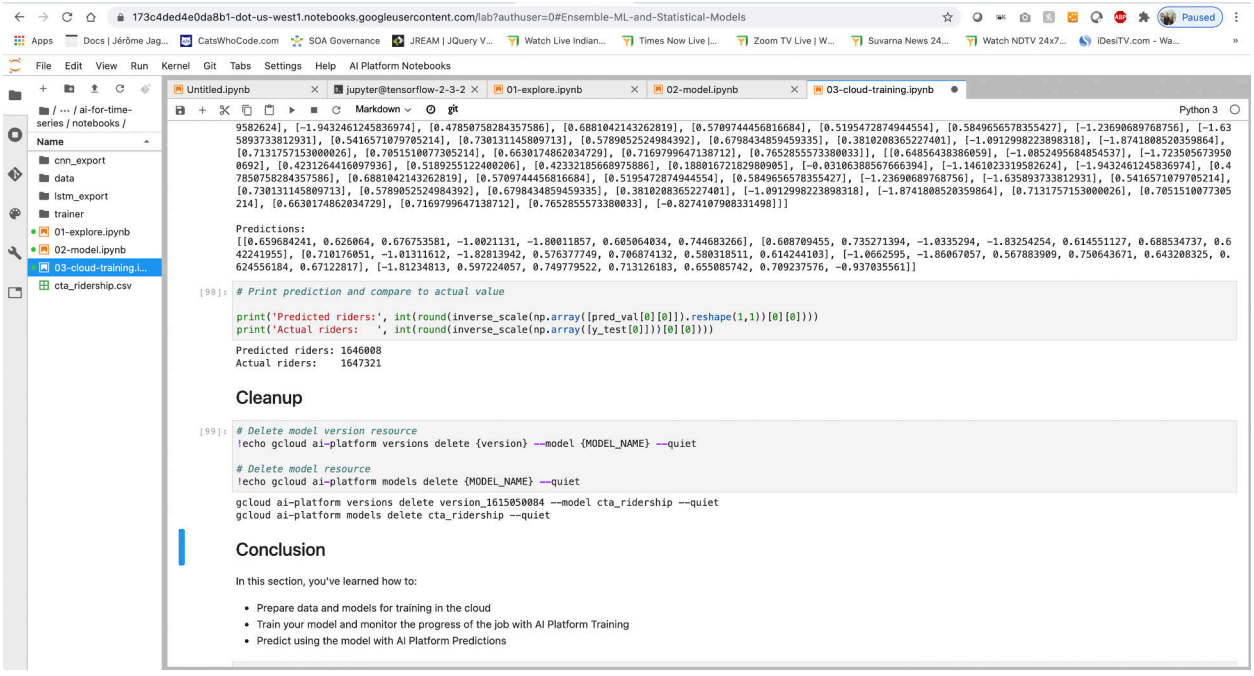

console.cloud.google.com/ai-platform/models/cta_ridership;region=us-central1/versions?project=legalpa-sandbox-4o9d
Google Cloud Platform
LegalPA
Search products and resources
Model Details
NEW VERSION
SHOW INFO PANEL
Name
cta_ridership
Default version
version_1615050084
Region
us-central1
Endpoint
us-central1-ml.googleapis.com
VERSIONS
EVALUATION
BETA
Filter
Filter by prefix...
Name
Create time
Last used
Labels
version_1615050084 (default)
Mar 6, 2021, 9:01:25 AM

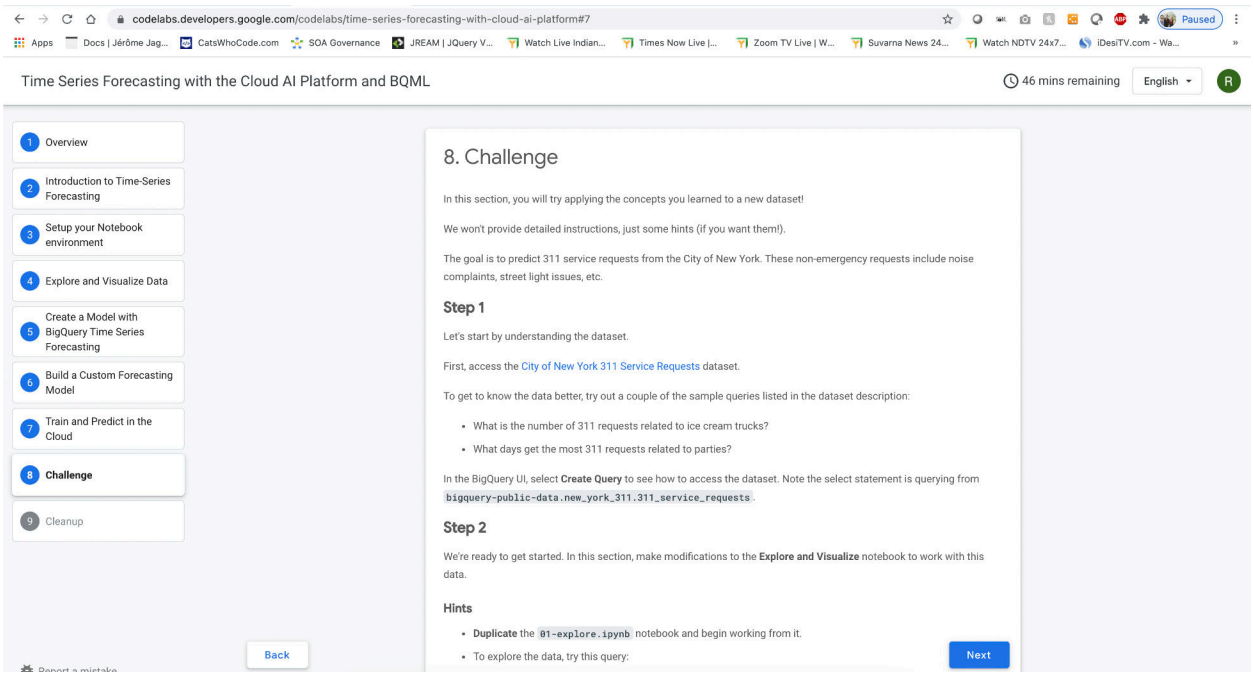


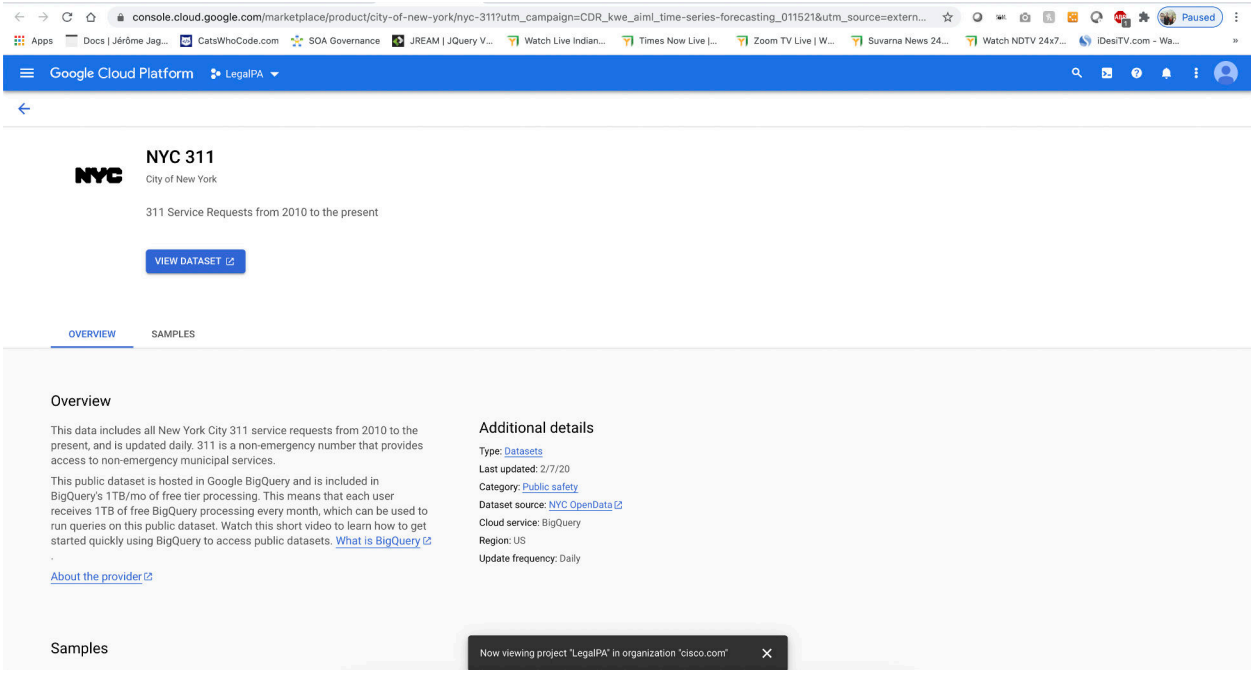
Prediction of the model

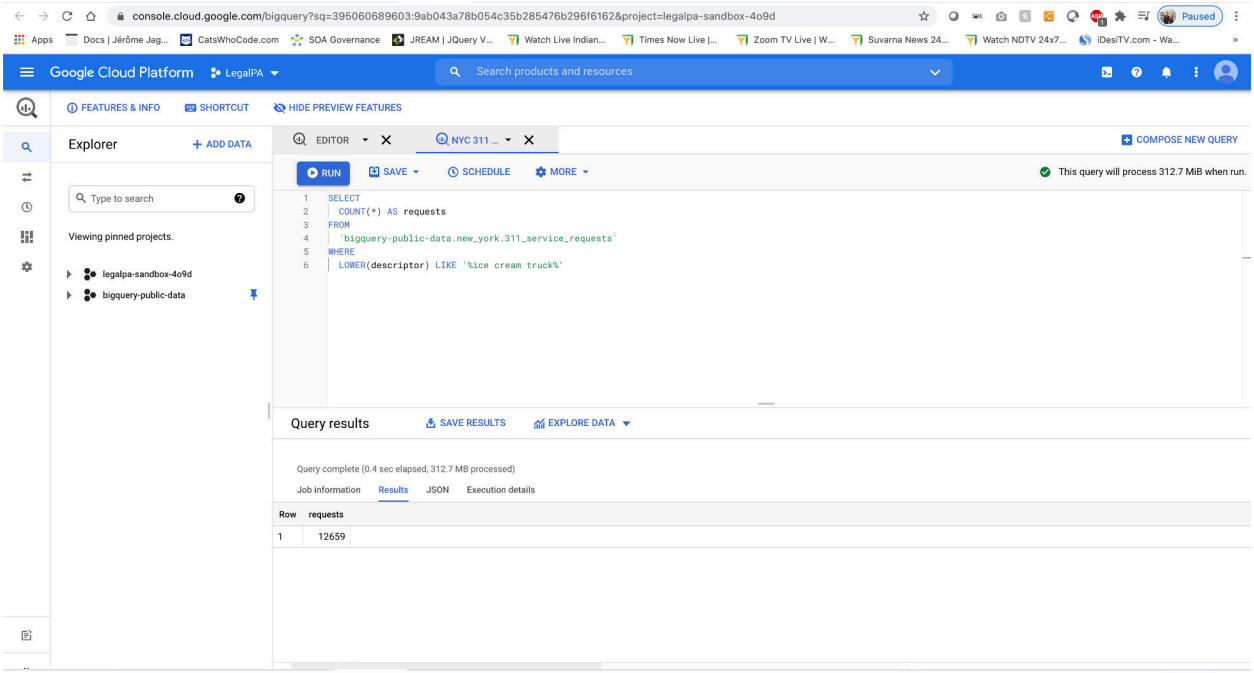


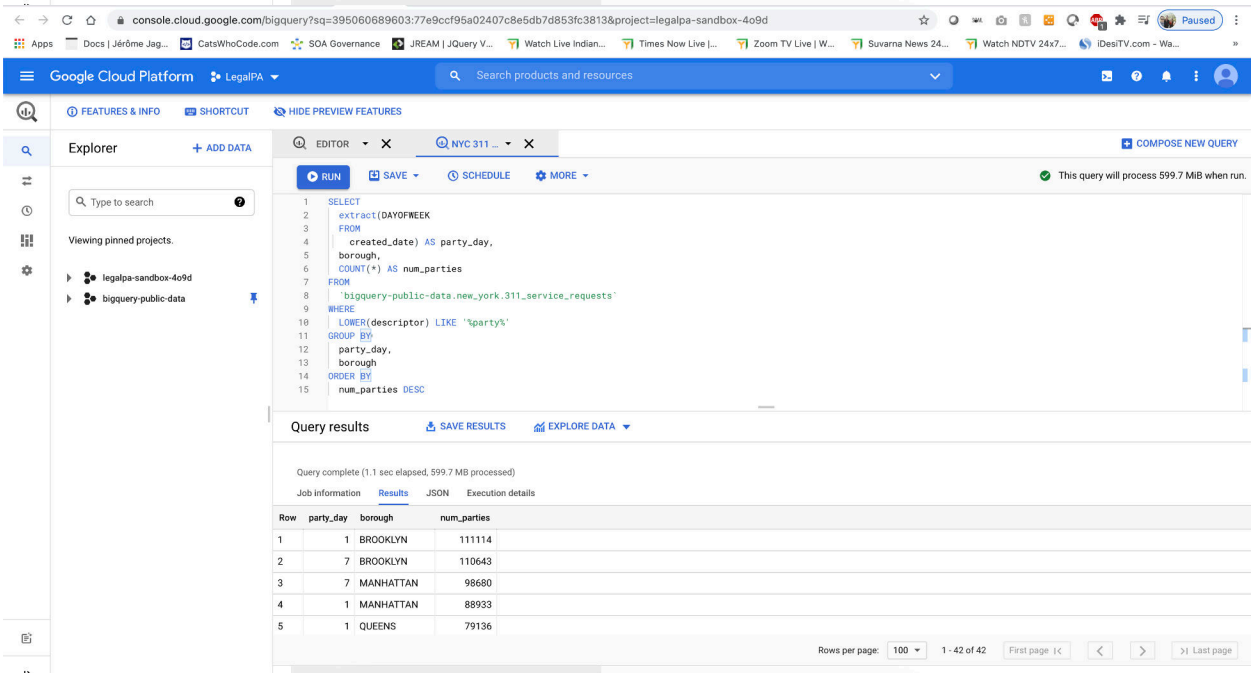

173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0#Ensemble-ML-and-Statistical-Models
Paused
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
Untitled.ipynb
jupyter@tensorflow-2-3-2
01-explore.ipynb
02-model.ipynb
03-cloud-training.ipynb
Code
Python 3
/ ··· / ai-for-time-series / notebooks /
Name
Last Modified
cnn_export 9 hours ago
data 10 hours ago
lstm_export 9 hours ago
trainer 8 hours ago
01-explore.ipynb 10 hours ago
02-model.ipynb 9 hours ago
03-cloud-training.ip... 3 minutes ago
cta_ridership.csv 10 hours ago
print(X_test.tolist()[:5])
pred_val = predict_json(PROJECT, REGION, MODEL_NAME, X_test.tolist(), version=version)
print()
print('Predictions:')
print(pred_val[:5])
Input:
Predictions:
[98]:
# Print prediction and compare to actual value
print('Predicted riders:', int(round(inverse_scale(np.array([pred_val[0][0]]).reshape(1,1))[0][0])))
print('Actual riders: ', int(round(inverse_scale(np.array([y_test[0]]))[0][0])))
Predicted riders: 1646008
Actual riders: 1647321

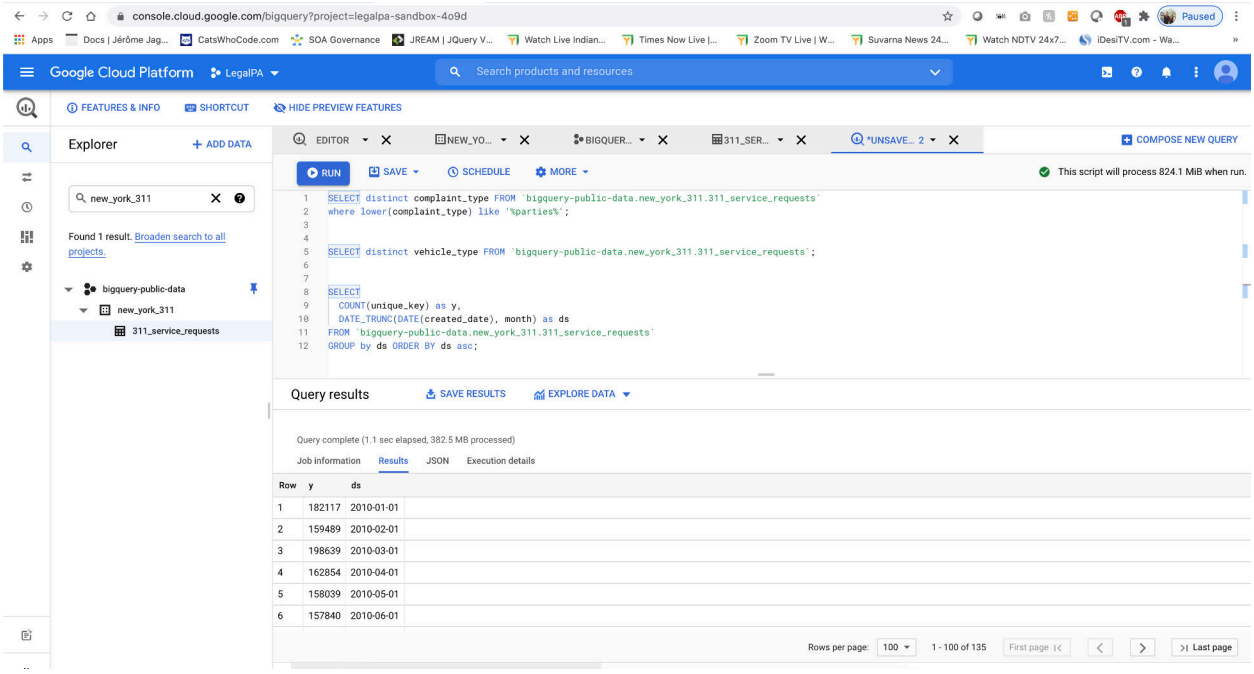

173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0#Ensemble-ML-and-Statistical-Models
Paused
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
Untitled.ipynb
jupyter@tensorflow-2-3-2
01-explore.ipynb
02-model.ipynb
03-cloud-training.ipynb
Markdown
Python 3
/ ··· / ai-for-time-series / notebooks /
Name
cnn_export
data
lstm_export
trainer
01-explore.ipynb
02-model.ipynb
03-cloud-training.i...
cta_ridership.csv
Predictions:
[98]:
# Print prediction and compare to actual value
print('Predicted riders:', int(round(inverse_scale(np.array([pred_val[0][0]]).reshape(1,1))[0][0])))
print('Actual riders: ', int(round(inverse_scale(np.array([y_test[0]]))[0][0])))
Predicted riders: 1646008
Actual riders: 1647321
Cleanup
[99]:
Delete model version resource
!echo gcloud ai-platform versions delete {version} --model {MODEL_NAME} --quiet
Delete model resource
!echo gcloud ai-platform models delete {MODEL_NAME} --quiet
gcloud ai-platform versions delete version_1615050084 --model cta_ridership --quiet
gcloud ai-platform models delete cta_ridership --quiet
Conclusion
In this section, you've learned how to:
Prepare data and models for training in the cloud
Train your model and monitor the progress of the job with AI Platform Training
Predict using the model with AI Platform Predictions

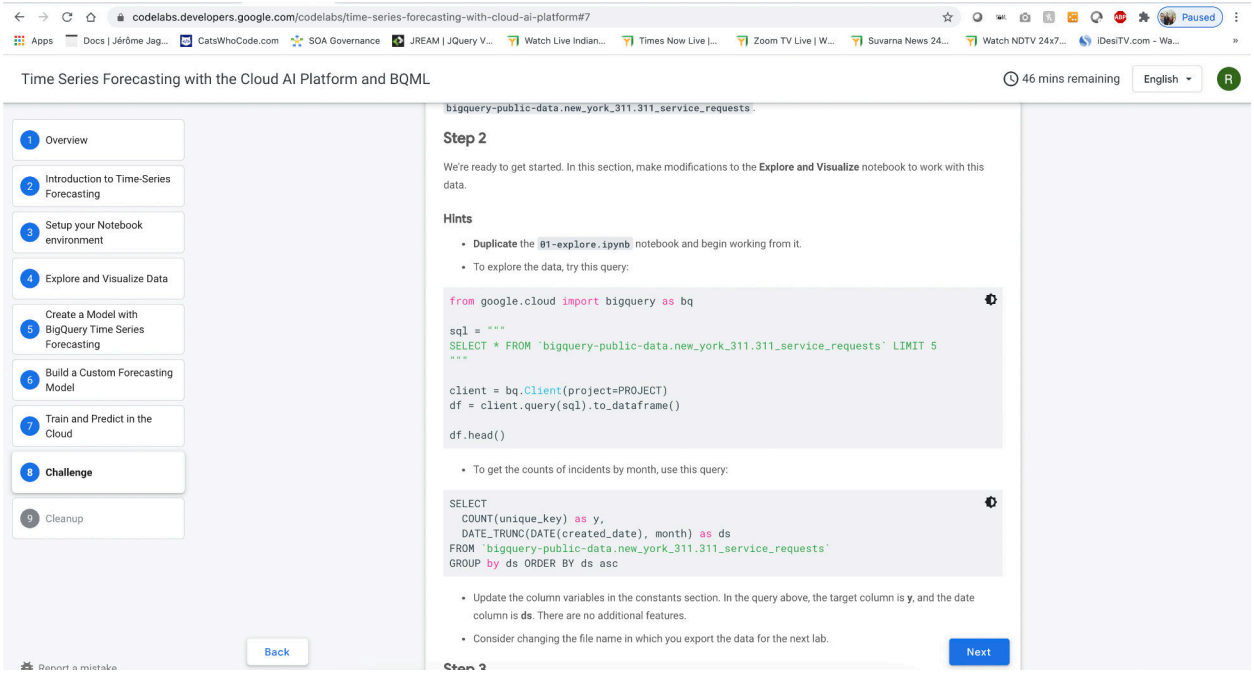

codelabs.developers.google.com/codelabs/time-series-forecasting-with-cloud-ai-platform#7
Time Series Forecasting with the Cloud AI Platform and BQML
46 mins remaining
English
1 Overview
2 Introduction to Time-Series Forecasting
3 Setup your Notebook environment
4 Explore and Visualize Data
5 Create a Model with BigQuery Time Series Forecasting
6 Build a Custom Forecasting Model
7 Train and Predict in the Cloud
8 Challenge
9 Cleanup
8. Challenge
In this section, you will try applying the concepts you learned to a new dataset!
We won't provide detailed instructions, just some hints (if you want them!).
The goal is to predict 311 service requests from the City of New York. These non-emergency requests include noise complaints, street light issues, etc.
Step 1
Let's start by understanding the dataset.
First, access the City of New York 311 Service Requests dataset.
To get to know the data better, try out a couple of the sample queries listed in the dataset description:
What is the number of 311 requests related to ice cream trucks?
What days get the most 311 requests related to parties?
In the BigQuery UI, select Create Query to see how to access the dataset. Note the select statement is querying from bigquery-public-data.new_york_311.311_service_requests.
Step 2
We're ready to get started. In this section, make modifications to the Explore and Visualize notebook to work with this data.
Hints
Duplicate the 01-explore.ipynb notebook and begin working from it.
To explore the data, try this query:
Back
Next

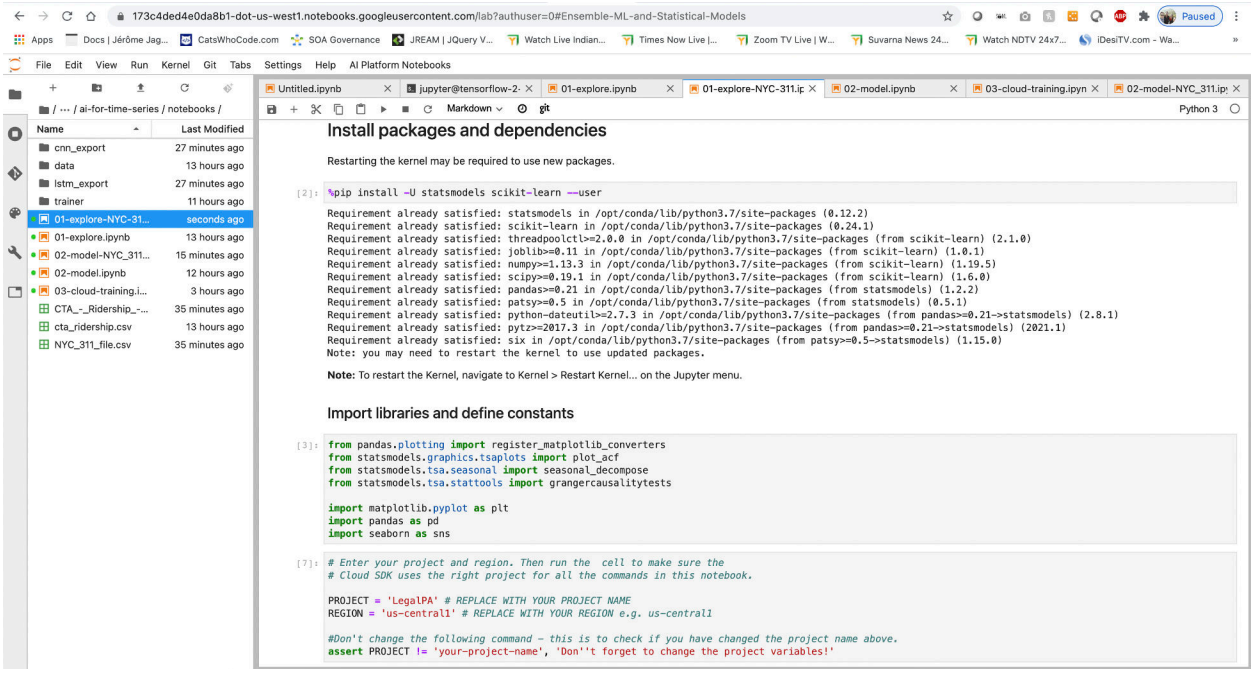

console.cloud.google.com/marketplace/product/city-of-new-york/nyc-311?utm_campaign=CDR_kwe_aiml_time-series-forecasting_011521&utm_source=extern...
Google Cloud Platform
LegalPA
NYC 311
City of New York
311 Service Requests from 2010 to the present
VIEW DATASET
OVERVIEW
SAMPLES
Overview
This data includes all New York City 311 service requests from 2010 to the present, and is updated daily. 311 is a non-emergency number that provides access to non-emergency municipal services.
This public dataset is hosted in Google BigQuery and is included in BigQuery's 1TB/mo of free tier processing. This means that each user receives 1TB of free BigQuery processing every month, which can be used to run queries on this public dataset. Watch this short video to learn how to get started quickly using BigQuery to access public datasets. What is BigQuery
About the provider
Additional details
Type: Datasets
Last updated: 2/7/20
Category: Public safety
Dataset source: NYC OpenData
Cloud service: BigQuery
Region: US
Update frequency: Daily
Samples
Now viewing project "LegalPA" in organization "cisco.com"

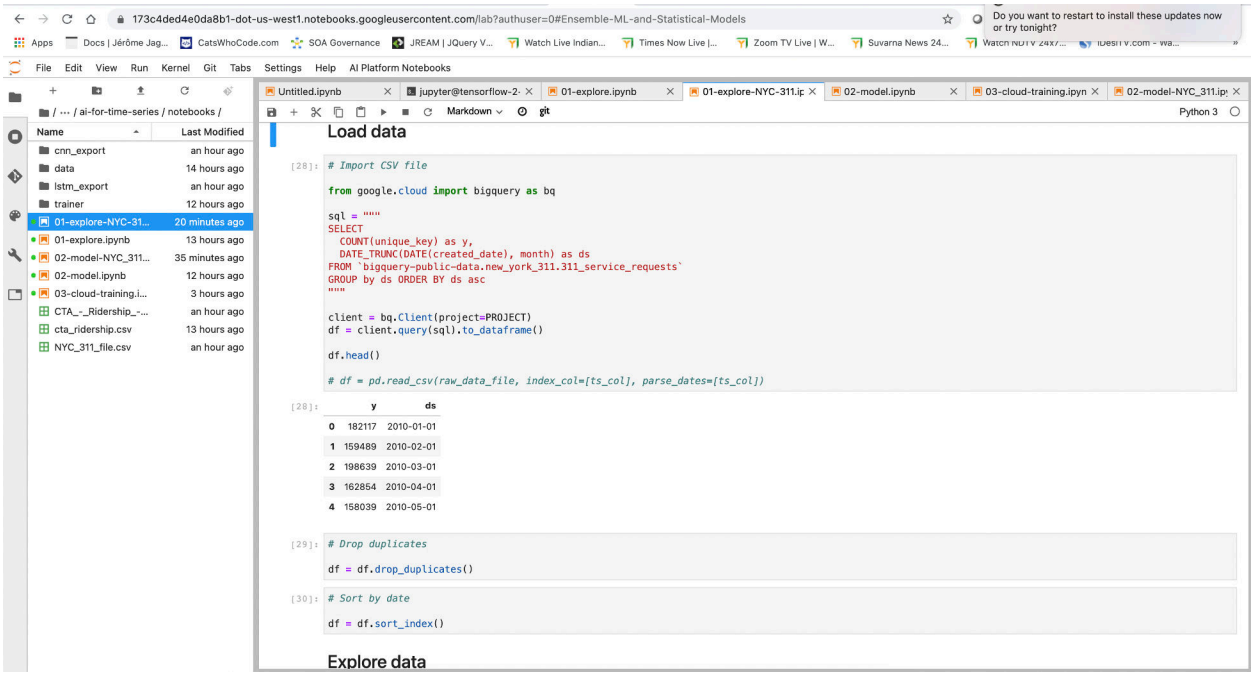

console.cloud.google.com/bigquery?sq=395060689603:9ab043a78b054c35b285476b296f6162&project=legalpa-sandbox-4o9d
Google Cloud Platform
LegalPA
Search products and resources
FEATURES & INFO
SHORTCUT
HIDE PREVIEW FEATURES
Explorer
ADD DATA
Type to search
Viewing pinned projects.
legalpa-sandbox-4o9d
bigquery-public-data
EDITOR
NYC 311 ...
COMPOSE NEW QUERY
RUN
SAVE
SCHEDULE
MORE
This query will process 312.7 MiB when run.
SELECT
COUNT(*) AS requests
FROM
`bigquery-public-data.new_york.311_service_requests`
WHERE
LOWER(descriptor) LIKE '%ice cream truck%'
Query results
SAVE RESULTS
EXPLORE DATA
Query complete (0.4 sec elapsed, 312.7 MB processed)
Job information
Results
JSON
Execution details
Row
requests
1
12659

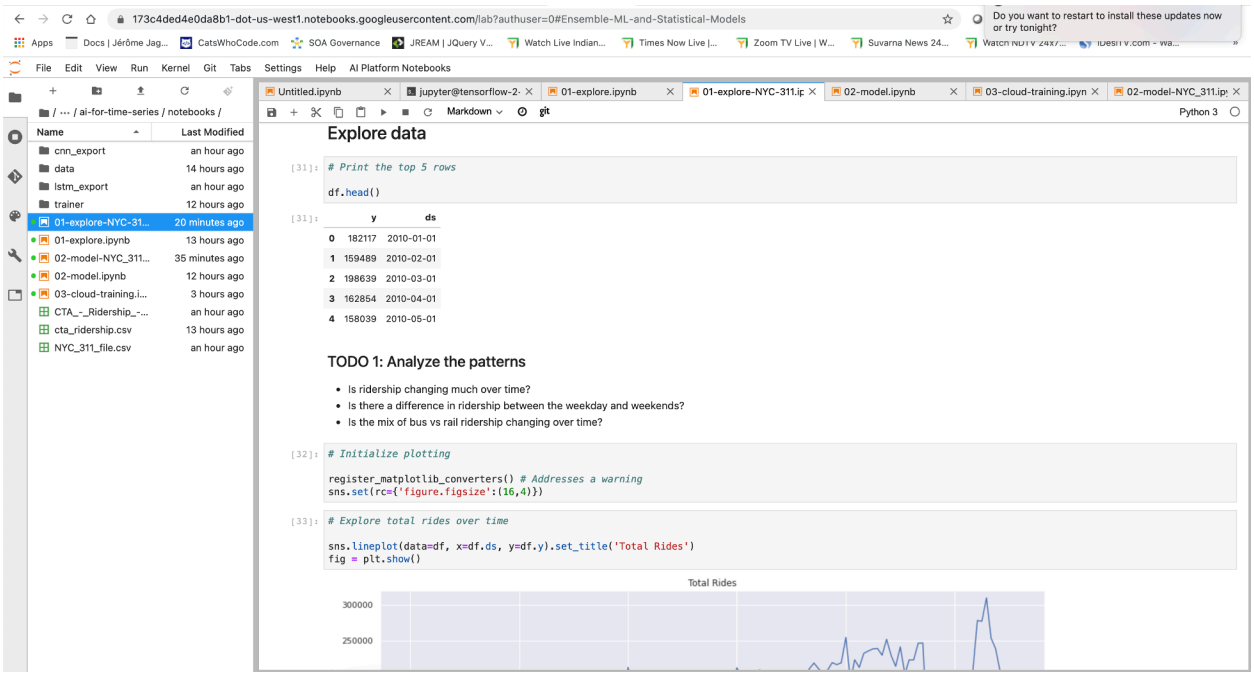

console.cloud.google.com/bigquery?sq=395060689603:77e9ccf95a02407c8e5db7d853fc3813&project=legalpa-sandbox-4o9d
Google Cloud Platform
LegalPA
Search products and resources
FEATURES & INFO
SHORTCUT
HIDE PREVIEW FEATURES
Explorer
ADD DATA
Type to search
Viewing pinned projects.
legalpa-sandbox-4o9d
bigquery-public-data
EDITOR
NYC 311 ...
COMPOSE NEW QUERY
RUN
SAVE
SCHEDULE
MORE
This query will process 599.7 MiB when run.
SELECT
extract(DAYOFWEEK
FROM
created_date) AS party_day,
borough,
COUNT(*) AS num_parties
FROM
`bigquery-public-data.new_york.311_service_requests`
WHERE
LOWER(descriptor) LIKE '%party%'
GROUP BY
party_day,
borough
ORDER BY
num_parties DESC
Query results
SAVE RESULTS
EXPLORE DATA
Query complete (1.1 sec elapsed, 599.7 MB processed)
Job information
Results
JSON
Execution details
Row
party_day
borough
num_parties
1
1
BROOKLYN
111114
2
7
BROOKLYN
110643
3
7
MANHATTAN
98680
4
1
MANHATTAN
88933
5
1
QUEENS
79136
Rows per page:
100
1 - 42 of 42
First page
Last page

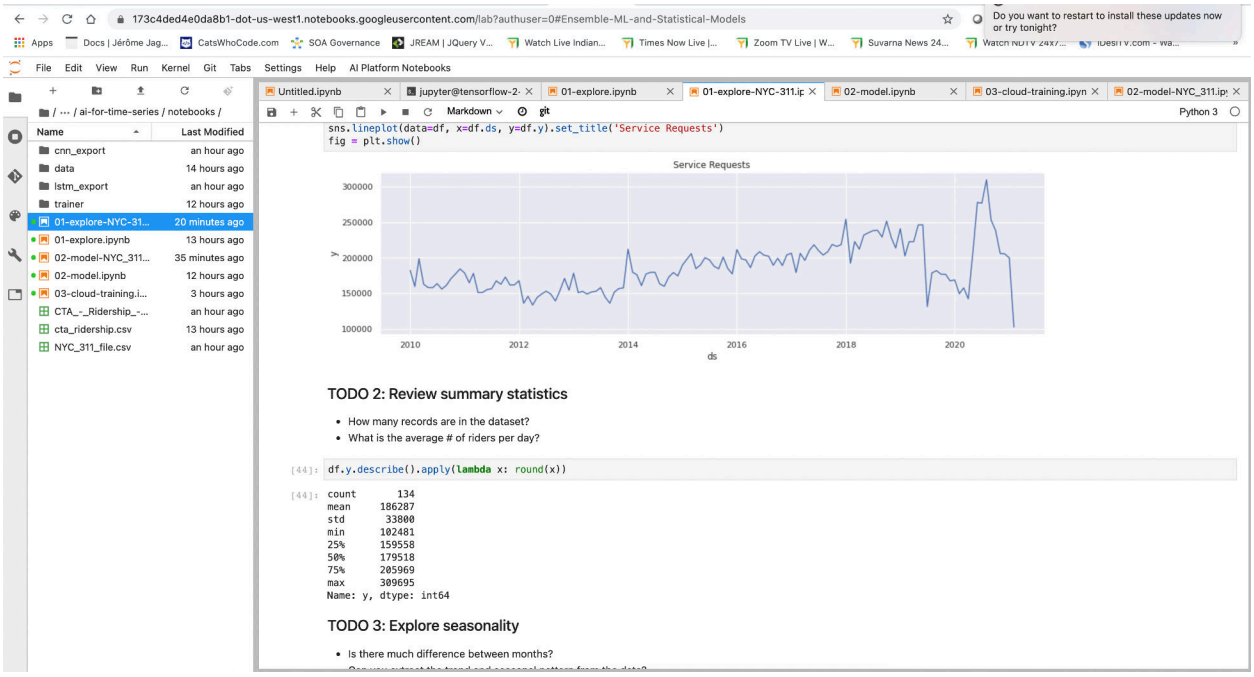

console.cloud.google.com/bigquery?project=legalpa-sandbox-4o9d
Google Cloud Platform
LegalPA
Search products and resources
FEATURES & INFO
SHORTCUT
HIDE PREVIEW FEATURES
Explorer
ADD DATA
new_york_311
Found 1 result. Broaden search to all projects.
bigquery-public-data
new_york_311
311_service_requests
EDITOR
COMPOSE NEW QUERY
RUN
SAVE
SCHEDULE
MORE
This script will process 824.1 MiB when run.
SELECT distinct complaint_type FROM `bigquery-public-data.new_york_311.311_service_requests`
where lower(complaint_type) like '%parties%';
SELECT distinct vehicle_type FROM `bigquery-public-data.new_york_311.311_service_requests`;
SELECT
COUNT(unique_key) as y,
DATE_TRUNC(DATE(created_date), month) as ds
FROM `bigquery-public-data.new_york_311.311_service_requests`
GROUP by ds ORDER BY ds asc;
Query results
SAVE RESULTS
EXPLORE DATA
Query complete (1.1 sec elapsed, 382.5 MB processed)
Job information
Results
JSON
Execution details
Row y ds
1 182117 2010-01-01
2 159489 2010-02-01
3 198639 2010-03-01
4 162854 2010-04-01
5 158039 2010-05-01
6 157840 2010-06-01
Rows per page: 100
1 - 100 of 135
First page
Last page

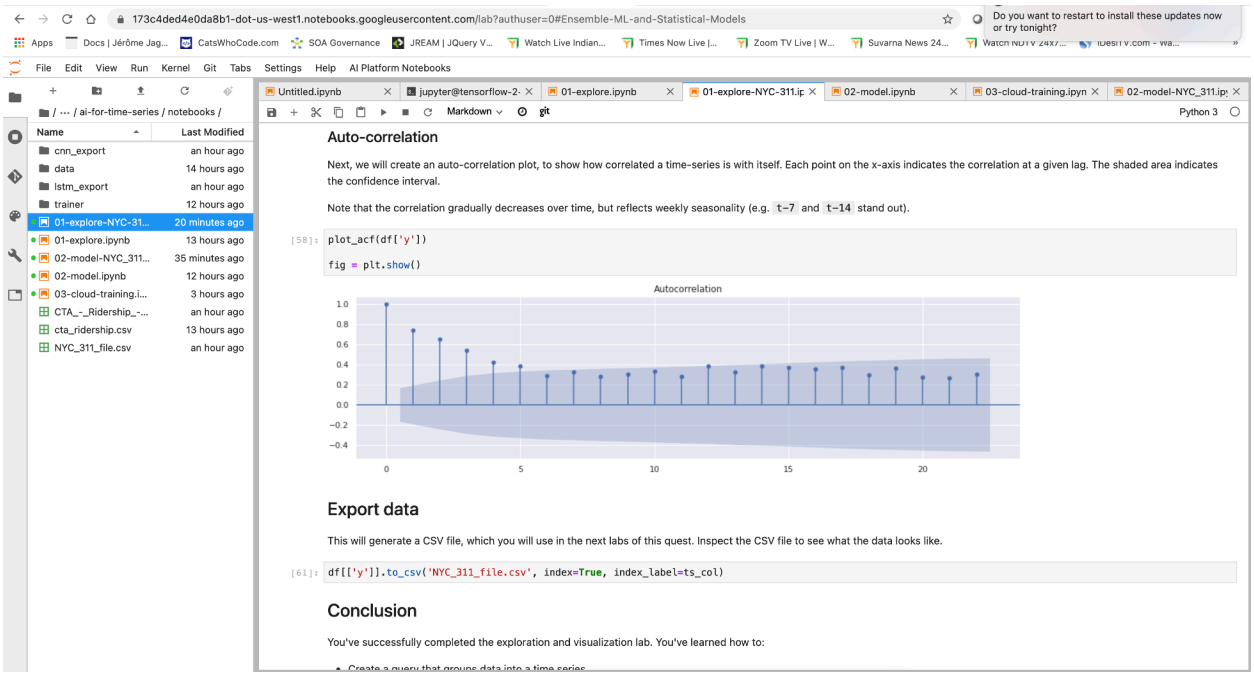

codelabs.developers.google.com/codelabs/time-series-forecasting-with-cloud-ai-platform#7
Time Series Forecasting with the Cloud AI Platform and BQML
46 mins remaining
English
1 Overview
2 Introduction to Time-Series Forecasting
3 Setup your Notebook environment
4 Explore and Visualize Data
5 Create a Model with BigQuery Time Series Forecasting
6 Build a Custom Forecasting Model
7 Train and Predict in the Cloud
8 Challenge
9 Cleanup
bigquery-public-data.new_york_311.311_service_requests.
Step 2
We're ready to get started. In this section, make modifications to the Explore and Visualize notebook to work with this data.
Hints
Duplicate the 01-explore.ipynb notebook and begin working from it.
To explore the data, try this query:
from google.cloud import bigquery as bq
sql = """
SELECT * FROM `bigquery-public-data.new_york_311.311_service_requests` LIMIT 5
"""
client = bq.Client(project=PROJECT)
df = client.query(sql).to_dataframe()
df.head()
To get the counts of incidents by month, use this query:
SELECT
COUNT(unique_key) as y,
DATE_TRUNC(DATE(created_date), month) as ds
FROM `bigquery-public-data.new_york_311.311_service_requests`
GROUP by ds ORDER BY ds asc
Update the column variables in the constants section. In the query above, the target column is y, and the date column is ds. There are no additional features.
Consider changing the file name in which you export the data for the next lab.
Back
Next

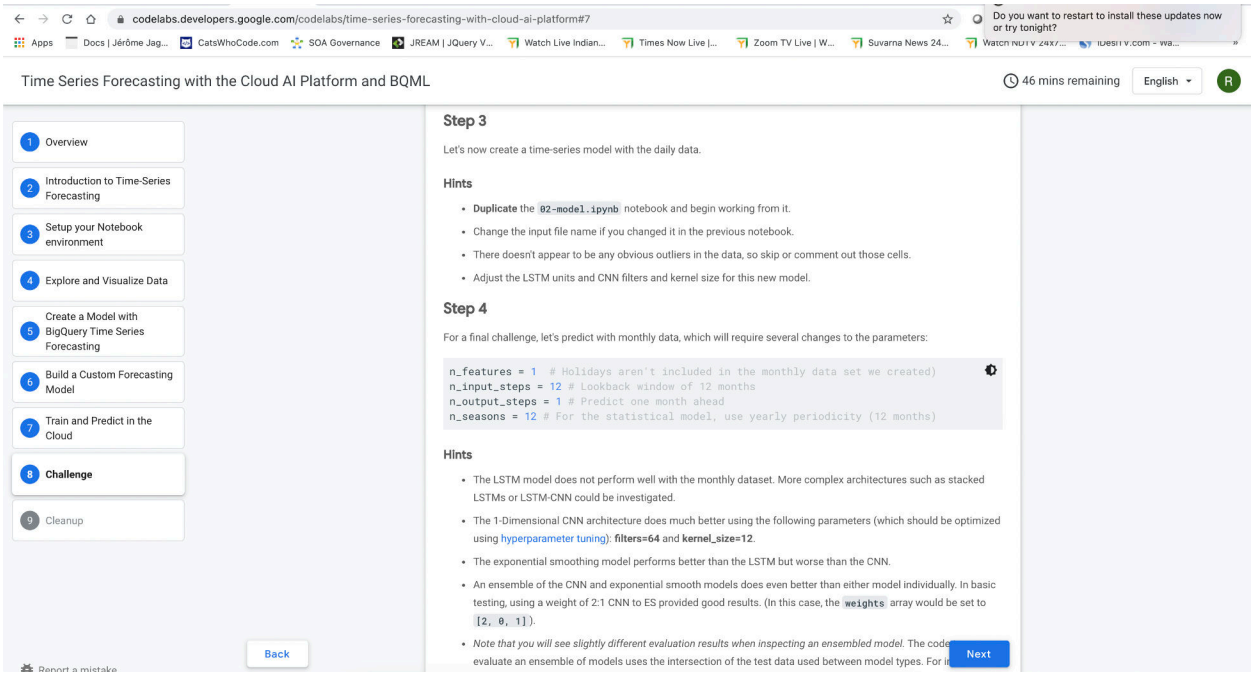

173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0#Ensemble-ML-and-Statistical-Models
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
Untitled.ipynb
jupyter@tensorflow-2·
01-explore.ipynb
01-explore-NYC-311.ip
02-model.ipynb
03-cloud-training.ipyn
02-model-NYC_311.ip
Markdown
git
Python 3
/ ··· / ai-for-time-series / notebooks /
Name
Last Modified
cnn_export 27 minutes ago
data 13 hours ago
lstm_export 27 minutes ago
trainer 11 hours ago
01-explore-NYC-31... seconds ago
01-explore.ipynb 13 hours ago
02-model-NYC_311... 15 minutes ago
02-model.ipynb 12 hours ago
03-cloud-training.i... 3 hours ago
CTA_-_Ridership_-... 35 minutes ago
cta_ridership.csv 13 hours ago
NYC_311_file.csv 35 minutes ago
Install packages and dependencies
Restarting the kernel may be required to use new packages.
[2]: %pip install -U statsmodels scikit-learn --user
Requirement already satisfied: statsmodels in /opt/conda/lib/python3.7/site-packages (0.12.2)
Requirement already satisfied: scikit-learn in /opt/conda/lib/python3.7/site-packages (0.24.1)
Requirement already satisfied: threadpoolctl>=2.0.0 in /opt/conda/lib/python3.7/site-packages (from scikit-learn) (2.1.0)
Requirement already satisfied: joblib>=0.11 in /opt/conda/lib/python3.7/site-packages (from scikit-learn) (1.0.1)
Requirement already satisfied: numpy>=1.13.3 in /opt/conda/lib/python3.7/site-packages (from scikit-learn) (1.19.5)
Requirement already satisfied: scipy>=0.19.1 in /opt/conda/lib/python3.7/site-packages (from scikit-learn) (1.6.0)
Requirement already satisfied: pandas>=0.21 in /opt/conda/lib/python3.7/site-packages (from statsmodels) (1.2.2)
Requirement already satisfied: patsy>=0.5 in /opt/conda/lib/python3.7/site-packages (from statsmodels) (0.5.1)
Requirement already satisfied: python-dateutil>=2.7.3 in /opt/conda/lib/python3.7/site-packages (from pandas>=0.21->statsmodels) (2.8.1)
Requirement already satisfied: pytz>=2017.3 in /opt/conda/lib/python3.7/site-packages (from pandas>=0.21->statsmodels) (2021.1)
Requirement already satisfied: six in /opt/conda/lib/python3.7/site-packages (from patsy>=0.5->statsmodels) (1.15.0)
Note: you may need to restart the kernel to use updated packages.
Note: To restart the Kernel, navigate to Kernel > Restart Kernel... on the Jupyter menu.
Import libraries and define constants
[3]: from pandas.plotting import register_matplotlib_converters
from statsmodels.graphics.tsaplots import plot_acf
from statsmodels.tsa.seasonal import seasonal_decompose
from statsmodels.tsa.stattools import grangercausalitytests
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
[7]: # Enter your project and region. Then run the cell to make sure the
# Cloud SDK uses the right project for all the commands in this notebook.
PROJECT = 'LegalPA' # REPLACE WITH YOUR PROJECT NAME
REGION = 'us-central1' # REPLACE WITH YOUR REGION e.g. us-central1
#Don't change the following command - this is to check if you have changed the project name above.
assert PROJECT != 'your-project-name', 'Don''t forget to change the project variables!'

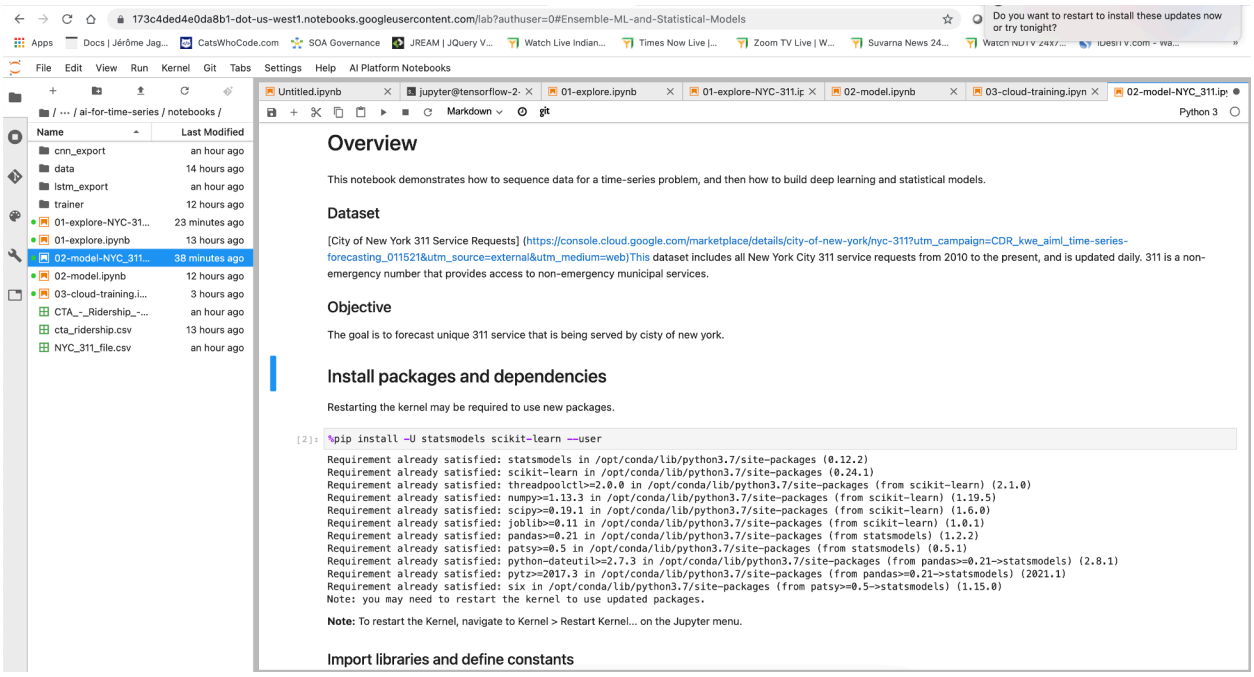

173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0#Ensemble-ML-and-Statistical-Models
Do you want to restart to install these updates now or try tonight?
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
Untitled.ipynb
jupyter@tensorflow-2·
01-explore.ipynb
01-explore-NYC-311.ip
02-model.ipynb
03-cloud-training.ipyn
02-model-NYC_311.ip
Markdown
git
Python 3
/ ··· / ai-for-time-series / notebooks /
Name
Last Modified
cnn_export an hour ago
data 14 hours ago
lstm_export an hour ago
trainer 12 hours ago
01-explore-NYC-31... 20 minutes ago
01-explore.ipynb 13 hours ago
02-model-NYC_311... 35 minutes ago
02-model.ipynb 12 hours ago
03-cloud-training.i... 3 hours ago
CTA_-_Ridership_-... an hour ago
cta_ridership.csv 13 hours ago
NYC_311_file.csv an hour ago
Load data
[28]: # Import CSV file
from google.cloud import bigquery as bq
sql = """
SELECT
COUNT(unique_key) as y,
DATE_TRUNC(DATE(created_date), month) as ds
FROM `bigquery-public-data.new_york_311.311_service_requests`
GROUP by ds ORDER BY ds asc
"""
client = bq.Client(project=PROJECT)
df = client.query(sql).to_dataframe()
df.head()
# df = pd.read_csv(raw_data_file, index_col=[ts_col], parse_dates=[ts_col])
[28]:
y ds
0 182117 2010-01-01
1 159489 2010-02-01
2 198639 2010-03-01
3 162854 2010-04-01
4 158039 2010-05-01
[29]: # Drop duplicates
df = df.drop_duplicates()
[30]: # Sort by date
df = df.sort_index()
Explore data

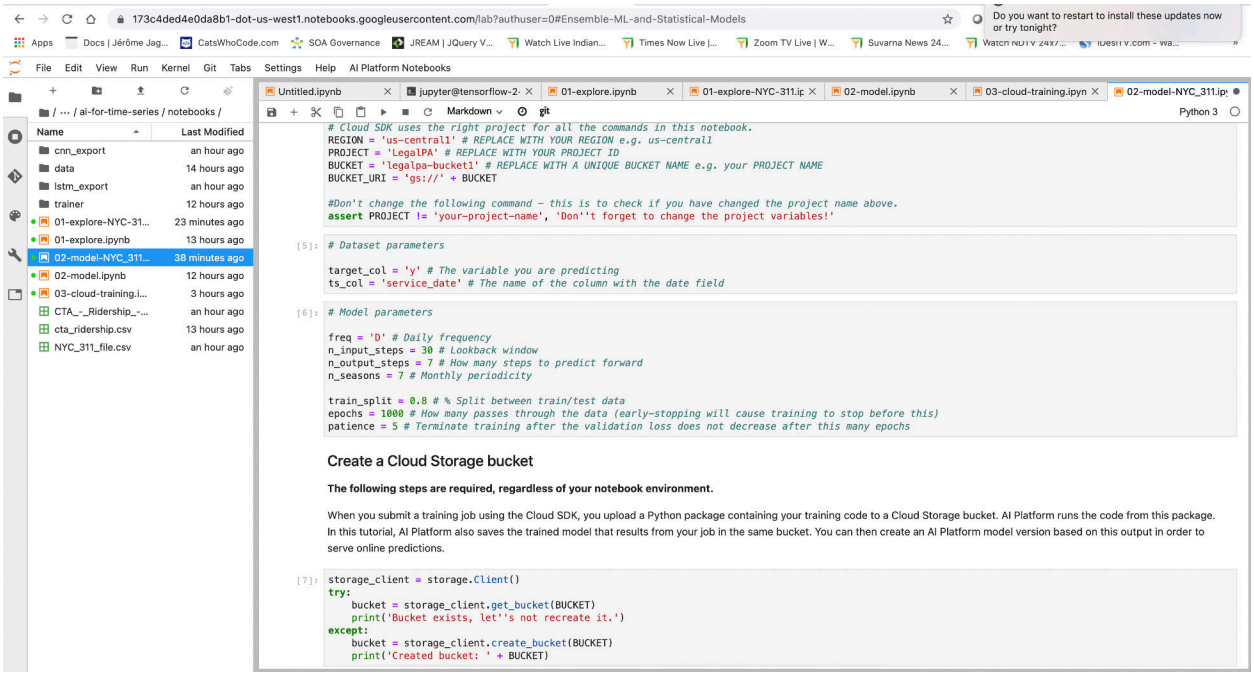

173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0#Ensemble-ML-and-Statistical-Models
Do you want to restart to install these updates now or try tonight?
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
Untitled.ipynb
jupyter@tensorflow-2·
01-explore.ipynb
01-explore-NYC-311.ip
02-model.ipynb
03-cloud-training.ipyn
02-model-NYC_311.ip
Markdown
git
Python 3
/ ··· / ai-for-time-series / notebooks /
Name
Last Modified
cnn_export an hour ago
data 14 hours ago
lstm_export an hour ago
trainer 12 hours ago
01-explore-NYC-31... 20 minutes ago
01-explore.ipynb 13 hours ago
02-model-NYC_311... 35 minutes ago
02-model.ipynb 12 hours ago
03-cloud-training.i... 3 hours ago
CTA_-_Ridership_-... an hour ago
cta_ridership.csv 13 hours ago
NYC_311_file.csv an hour ago
Explore data
[31]:
# Print the top 5 rows
df.head()
[31]:
y ds
0 182117 2010-01-01
1 159489 2010-02-01
2 198639 2010-03-01
3 162854 2010-04-01
4 158039 2010-05-01
TODO 1: Analyze the patterns
Is ridership changing much over time?
Is there a difference in ridership between the weekday and weekends?
Is the mix of bus vs rail ridership changing over time?
[32]:
Initialize plotting
register_matplotlib_converters() # Addresses a warning
sns.set(rc={'figure.figsize':(16,4)})
[33]:
# Explore total rides over time
sns.lineplot(data=df, x=df.ds, y=df.y).set_title('Total Rides')
fig = plt.show()
Total Rides
300000
250000

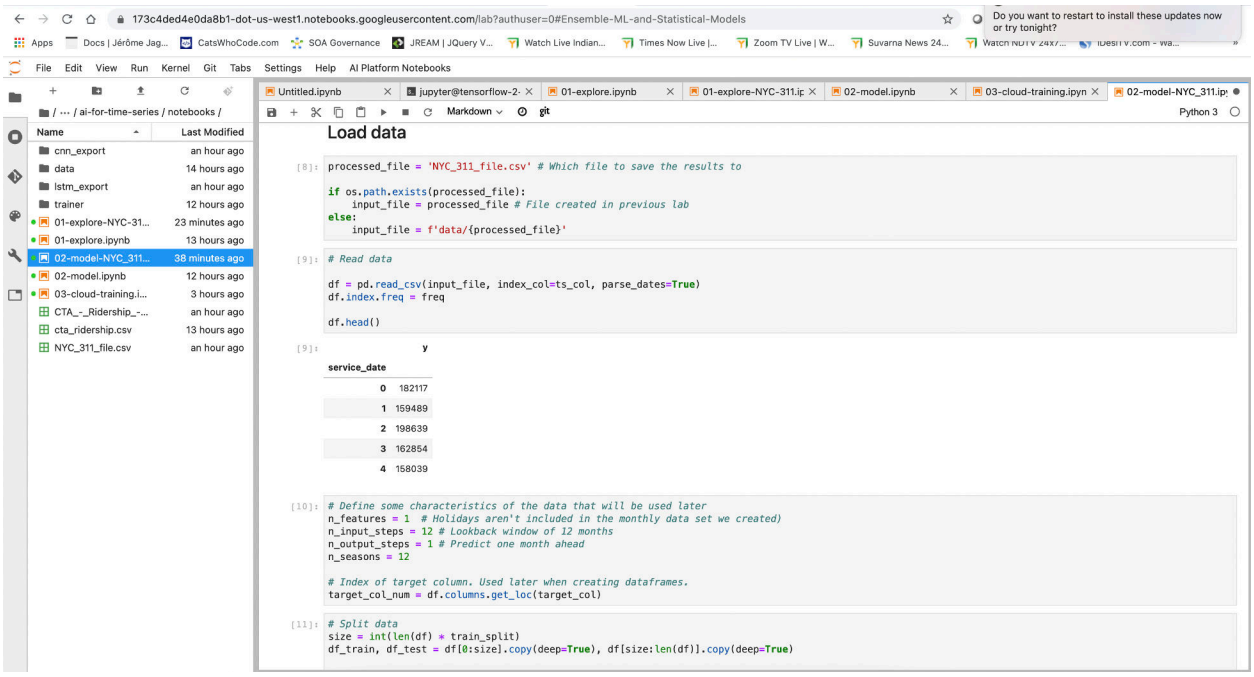

173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0#Ensemble-ML-and-Statistical-Models
Do you want to restart to install these updates now or try tonight?
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
Untitled.ipynb
jupyter@tensorflow-2·
01-explore.ipynb
01-explore-NYC-311.ip
02-model.ipynb
03-cloud-training.ipyn
02-model-NYC_311.ip
Markdown
git
Python 3
/ ··· / ai-for-time-series / notebooks /
Name
Last Modified
cnn_export an hour ago
data 14 hours ago
lstm_export an hour ago
trainer 12 hours ago
01-explore-NYC-31... 20 minutes ago
01-explore.ipynb 13 hours ago
02-model-NYC_311... 35 minutes ago
02-model.ipynb 12 hours ago
03-cloud-training.i... 3 hours ago
CTA_-_Ridership_-... an hour ago
cta_ridership.csv 13 hours ago
NYC_311_file.csv an hour ago
sns.lineplot(data=df, x=df.ds, y=df.y).set_title('Service Requests')
fig = plt.show()
Service Requests
300000
250000
200000
150000
100000
y
2010
2012
2014
2016
2018
2020
ds
TODO 2: Review summary statistics
How many records are in the dataset?
What is the average # of riders per day?
[44]:
df.y.describe().apply(lambda x: round(x))
[44]:
count 134
mean 186287
std 33800
min 102481
25% 159558
50% 179518
75% 205969
max 309695
Name: y, dtype: int64
TODO 3: Explore seasonality
Is there much difference between months?

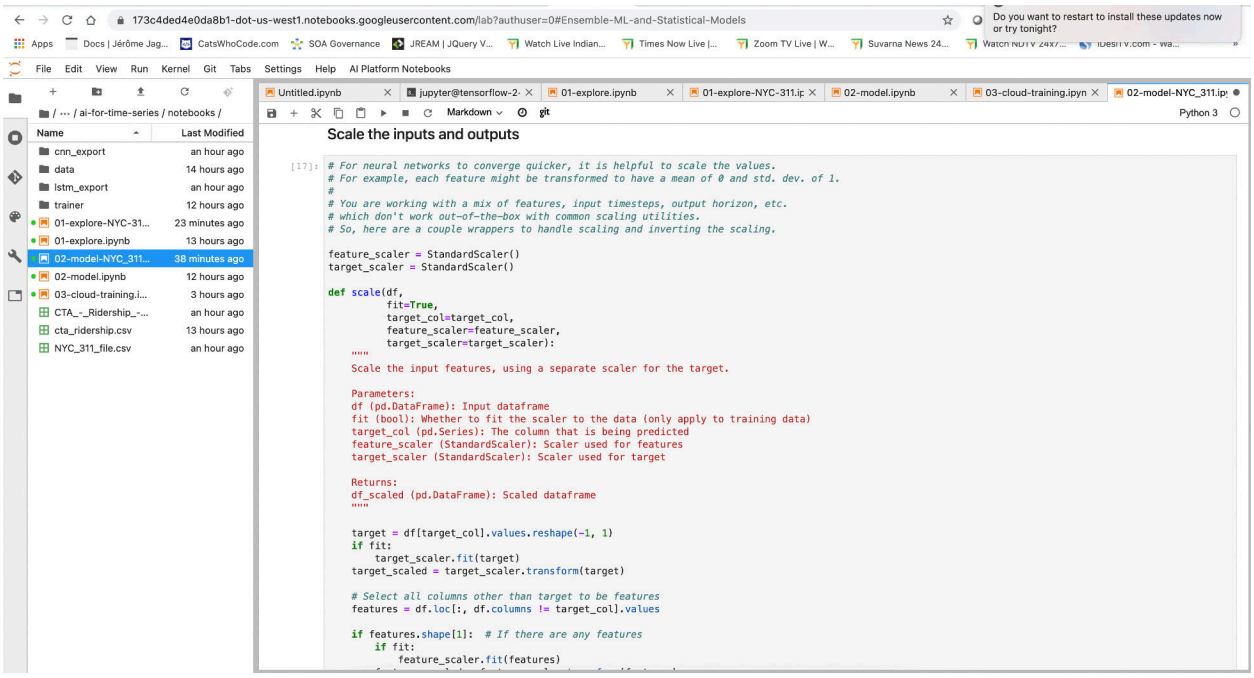

173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0#Ensemble-ML-and-Statistical-Models
Do you want to restart to install these updates now or try tonight?
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
Untitled.ipynb
jupyter@tensorflow-2-
01-explore.ipynb
01-explore-NYC-311.ip
02-model.ipynb
03-cloud-training.ipyn
02-model-NYC_311.ip
Markdown
Python 3
/ ··· / ai-for-time-series / notebooks /
Name
Last Modified
cnn_export an hour ago
data 14 hours ago
lstm_export an hour ago
trainer 12 hours ago
01-explore-NYC-31... 20 minutes ago
01-explore.ipynb 13 hours ago
02-model-NYC_311... 35 minutes ago
02-model.ipynb 12 hours ago
03-cloud-training.i... 3 hours ago
CTA_-_Ridership_-... an hour ago
cta_ridership.csv 13 hours ago
NYC_311_file.csv an hour ago
Auto-correlation
Next, we will create an auto-correlation plot, to show how correlated a time-series is with itself. Each point on the x-axis indicates the correlation at a given lag. The shaded area indicates the confidence interval.
Note that the correlation gradually decreases over time, but reflects weekly seasonality (e.g. t-7 and t-14 stand out).
[58]: plot_acf(df['y'])
fig = plt.show()
Autocorrelation
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
0
5
10
15
20
Export data
This will generate a CSV file, which you will use in the next labs of this quest. Inspect the CSV file to see what the data looks like.
[61]: df[['y']].to_csv('NYC_311_file.csv', index=True, index_label=ts_col)
Conclusion
You've successfully completed the exploration and visualization lab. You've learned how to:

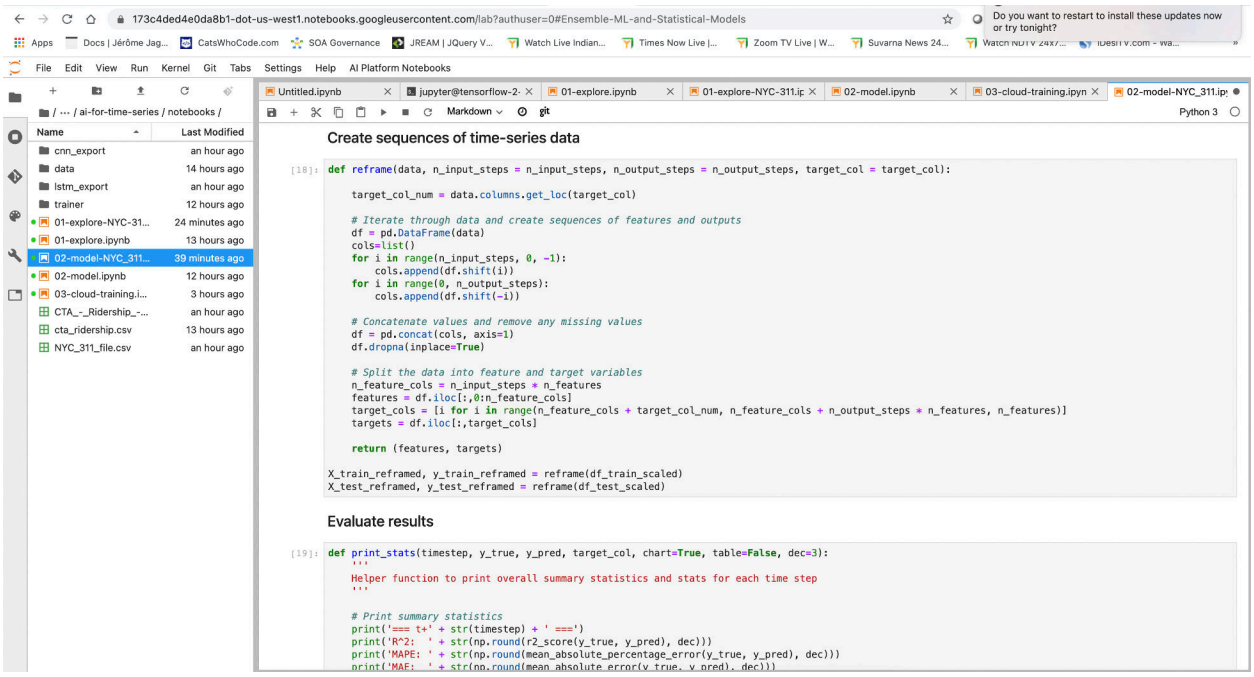

codelabs.developers.google.com/codelabs/time-series-forecasting-with-cloud-ai-platform#7
Do you want to restart to install these updates now or try tonight?
Time Series Forecasting with the Cloud AI Platform and BQML
46 mins remaining
English
1 Overview
2 Introduction to Time-Series Forecasting
3 Setup your Notebook environment
4 Explore and Visualize Data
5 Create a Model with BigQuery Time Series Forecasting
6 Build a Custom Forecasting Model
7 Train and Predict in the Cloud
8 Challenge
9 Cleanup
Step 3
Let's now create a time-series model with the daily data.
Hints
Duplicate the 02-model.ipynb notebook and begin working from it.
Change the input file name if you changed it in the previous notebook.
There doesn't appear to be any obvious outliers in the data, so skip or comment out those cells.
Adjust the LSTM units and CNN filters and kernel size for this new model.
Step 4
For a final challenge, let's predict with monthly data, which will require several changes to the parameters:
n_features = 1 # Holidays aren't included in the monthly data set we created)
n_input_steps = 12 # Lookback window of 12 months
n_output_steps = 1 # Predict one month ahead
n_seasons = 12 # For the statistical model, use yearly periodicity (12 months)
Hints
The LSTM model does not perform well with the monthly dataset. More complex architectures such as stacked LSTMs or LSTM-CNN could be investigated.
The 1-Dimensional CNN architecture does much better using the following parameters (which should be optimized using hyperparameter tuning): filters=64 and kernel_size=12.
The exponential smoothing model performs better than the LSTM but worse than the CNN.
An ensemble of the CNN and exponential smooth models does even better than either model individually. In basic testing, using a weight of 2:1 CNN to ES provided good results. (In this case, the weights array would be set to [2, 0, 1]).
Note that you will see slightly different evaluation results when inspecting an ensembled model. The code
evaluate an ensemble of models uses the intersection of the test data used between model types. For
Back
Next
Report a mistake

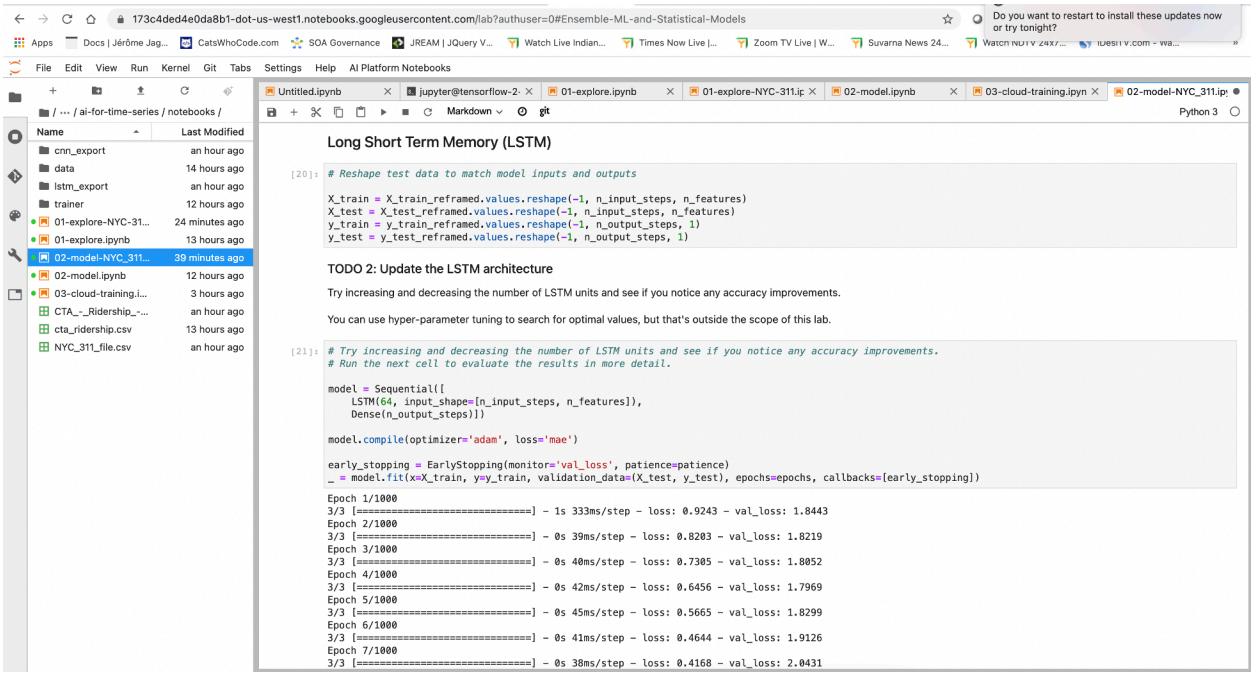

173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0#Ensemble-ML-and-Statistical-Models
Do you want to restart to install these updates now or try tonight?
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
/ ··· / ai-for-time-series / notebooks /
Name
Last Modified
cnn_export an hour ago
data 14 hours ago
lstm_export an hour ago
trainer 12 hours ago
01-explore-NYC-31... 23 minutes ago
01-explore.ipynb 13 hours ago
02-model-NYC_311... 38 minutes ago
02-model.ipynb 12 hours ago
03-cloud-training.i... 3 hours ago
CTA_-_Ridership_-... an hour ago
cta_ridership.csv 13 hours ago
NYC_311_file.csv an hour ago
Untitled.ipynb
jupyter@tensorflow-2·
01-explore.ipynb
01-explore-NYC-311.ip
02-model.ipynb
03-cloud-training.ipyn
02-model-NYC_311.ip
Markdown
git
Python 3
Overview
This notebook demonstrates how to sequence data for a time-series problem, and then how to build deep learning and statistical models.
Dataset
[City of New York 311 Service Requests] (https://console.cloud.google.com/marketplace/details/city-of-new-york/nyc-311?utm_campaign=CDR_kwe_aiml_time-series-forecasting_011521&utm_source=external&utm_medium=web)This dataset includes all New York City 311 service requests from 2010 to the present, and is updated daily. 311 is a non-emergency number that provides access to non-emergency municipal services.
Objective
The goal is to forecast unique 311 service that is being served by cisty of new york.
Install packages and dependencies
Restarting the kernel may be required to use new packages.
[2]:
%pip install -U statsmodels scikit-learn --user
Requirement already satisfied: statsmodels in /opt/conda/lib/python3.7/site-packages (0.12.2)
Requirement already satisfied: scikit-learn in /opt/conda/lib/python3.7/site-packages (0.24.1)
Requirement already satisfied: threadpoolctl>=2.0.0 in /opt/conda/lib/python3.7/site-packages (from scikit-learn) (2.1.0)
Requirement already satisfied: numpy>=1.13.3 in /opt/conda/lib/python3.7/site-packages (from scikit-learn) (1.19.5)
Requirement already satisfied: scipy>=0.19.1 in /opt/conda/lib/python3.7/site-packages (from scikit-learn) (1.6.0)
Requirement already satisfied: joblib>=0.11 in /opt/conda/lib/python3.7/site-packages (from scikit-learn) (1.0.1)
Requirement already satisfied: pandas>=0.21 in /opt/conda/lib/python3.7/site-packages (from statsmodels) (1.2.2)
Requirement already satisfied: patsy>=0.5 in /opt/conda/lib/python3.7/site-packages (from statsmodels) (0.5.1)
Requirement already satisfied: python-dateutil>=2.7.3 in /opt/conda/lib/python3.7/site-packages (from pandas>=0.21->statsmodels) (2.8.1)
Requirement already satisfied: pytz>=2017.3 in /opt/conda/lib/python3.7/site-packages (from pandas>=0.21->statsmodels) (2021.1)
Requirement already satisfied: six in /opt/conda/lib/python3.7/site-packages (from patsy>=0.5->statsmodels) (1.15.0)
Note: you may need to restart the kernel to use updated packages.
Note: To restart the Kernel, navigate to Kernel > Restart Kernel... on the Jupyter menu.
Import libraries and define constants

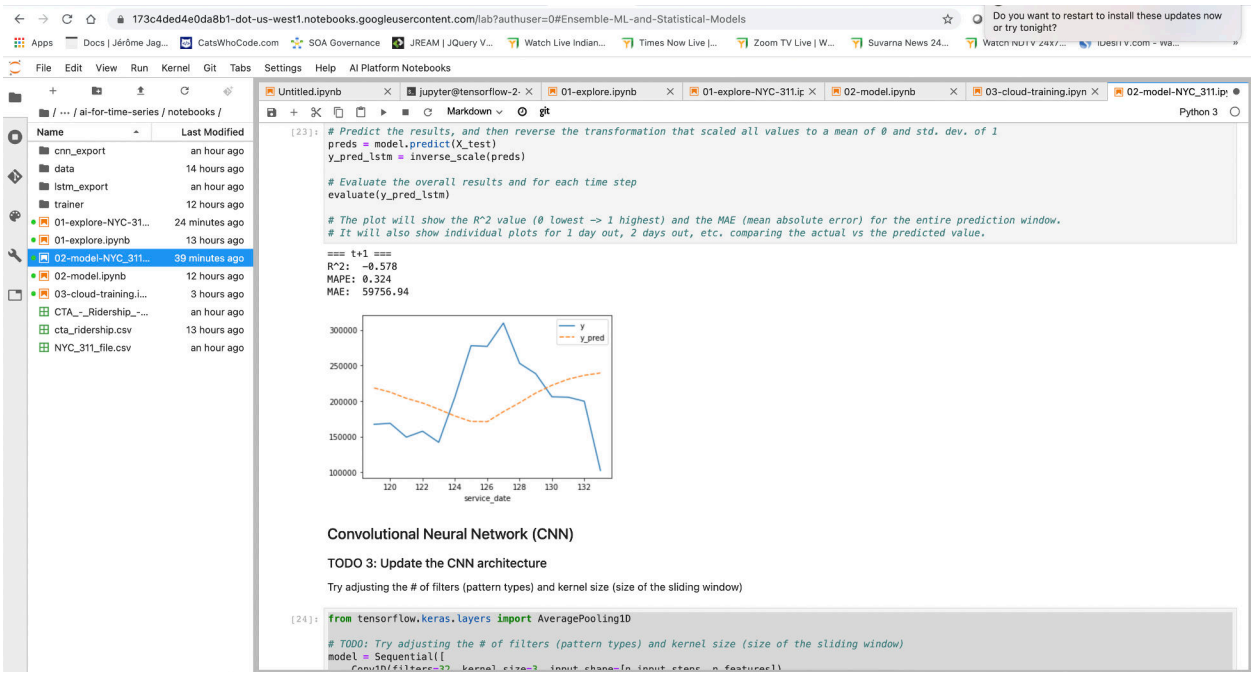

173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0#Ensemble-ML-and-Statistical-Models
Do you want to restart to install these updates now or try tonight?
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
/ ··· / ai-for-time-series / notebooks /
Name
Last Modified
cnn_export an hour ago
data 14 hours ago
lstm_export an hour ago
trainer 12 hours ago
01-explore-NYC-31... 23 minutes ago
01-explore.ipynb 13 hours ago
02-model-NYC_311... 38 minutes ago
02-model.ipynb 12 hours ago
03-cloud-training.i... 3 hours ago
CTA_-_Ridership_-... an hour ago
cta_ridership.csv 13 hours ago
NYC_311_file.csv an hour ago
Untitled.ipynb
jupyter@tensorflow-2·
01-explore.ipynb
01-explore-NYC-311.ip
02-model.ipynb
03-cloud-training.ipyn
02-model-NYC_311.ip
Markdown
git
Python 3
# Cloud SDK uses the right project for all the commands in this notebook.
REGION = 'us-central1' # REPLACE WITH YOUR REGION e.g. us-central1
PROJECT = 'LegalPA' # REPLACE WITH YOUR PROJECT ID
BUCKET = 'legalpa-bucket1' # REPLACE WITH A UNIQUE BUCKET NAME e.g. your PROJECT NAME
BUCKET_URI = 'gs://' + BUCKET
#Don't change the following command - this is to check if you have changed the project name above.
assert PROJECT != 'your-project-name', 'Don''t forget to change the project variables!'
[5]:
# Dataset parameters
target_col = 'y' # The variable you are predicting
ts_col = 'service_date' # The name of the column with the date field
[6]:
# Model parameters
freq = 'D' # Daily frequency
n_input_steps = 30 # Lookback window
n_output_steps = 7 # How many steps to predict forward
n_seasons = 7 # Monthly periodicity
train_split = 0.8 # % Split between train/test data
epochs = 1000 # How many passes through the data (early-stopping will cause training to stop before this)
patience = 5 # Terminate training after the validation loss does not decrease after this many epochs
Create a Cloud Storage bucket
The following steps are required, regardless of your notebook environment.
When you submit a training job using the Cloud SDK, you upload a Python package containing your training code to a Cloud Storage bucket. AI Platform runs the code from this package. In this tutorial, AI Platform also saves the trained model that results from your job in the same bucket. You can then create an AI Platform model version based on this output in order to serve online predictions.
[7]:
storage_client = storage.Client()
try:
bucket = storage_client.get_bucket(BUCKET)
print('Bucket exists, let''s not recreate it.')
except:
bucket = storage_client.create_bucket(BUCKET)
print('Created bucket: ' + BUCKET)

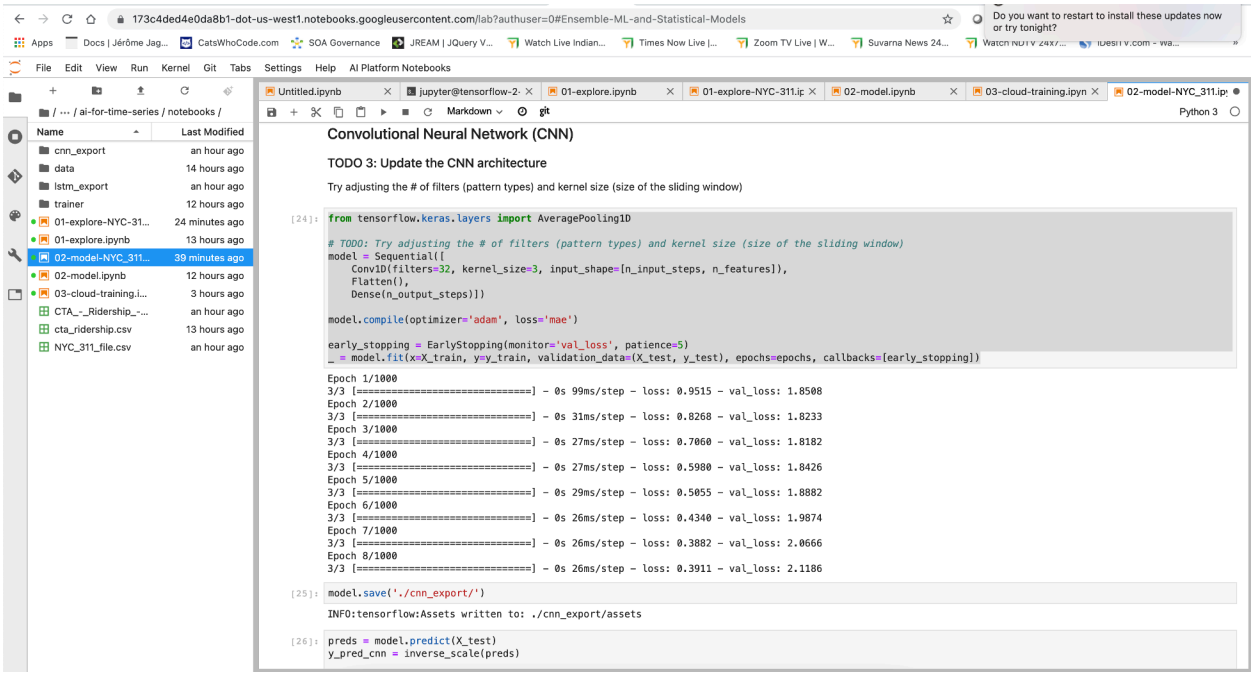

173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0#Ensemble-ML-and-Statistical-Models
Do you want to restart to install these updates now or try tonight?
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
Untitled.ipynb
jupyter@tensorflow-2·
01-explore.ipynb
01-explore-NYC-311.ip
02-model.ipynb
03-cloud-training.ipyn
02-model-NYC_311.ip
Markdown
git
Python 3
/ ··· / ai-for-time-series / notebooks /
Name
Last Modified
cnn_export an hour ago
data 14 hours ago
lstm_export an hour ago
trainer 12 hours ago
01-explore-NYC-31... 23 minutes ago
01-explore.ipynb 13 hours ago
02-model-NYC_311... 38 minutes ago
02-model.ipynb 12 hours ago
03-cloud-training.i... 3 hours ago
CTA_-_Ridership_-... an hour ago
cta_ridership.csv 13 hours ago
NYC_311_file.csv an hour ago
Load data
[8]:
processed_file = 'NYC_311_file.csv' # Which file to save the results to
if os.path.exists(processed_file):
input_file = processed_file # File created in previous lab
else:
input_file = f'data/{processed_file}'
[9]:
# Read data
df = pd.read_csv(input_file, index_col=ts_col, parse_dates=True)
df.index.freq = freq
df.head()
[9]:
y
service_date
0 182117
1 159489
2 198639
3 162854
4 158039
[10]:
# Define some characteristics of the data that will be used later
n_features = 1 # Holidays aren't included in the monthly data set we created)
n_input_steps = 12 # Lookback window of 12 months
n_output_steps = 1 # Predict one month ahead
n_seasons = 12
# Index of target column. Used later when creating dataframes.
target_col_num = df.columns.get_loc(target_col)
[11]:
# Split data
size = int(len(df) * train_split)
df_train, df_test = df[0:size].copy(deep=True), df[size:len(df)].copy(deep=True)

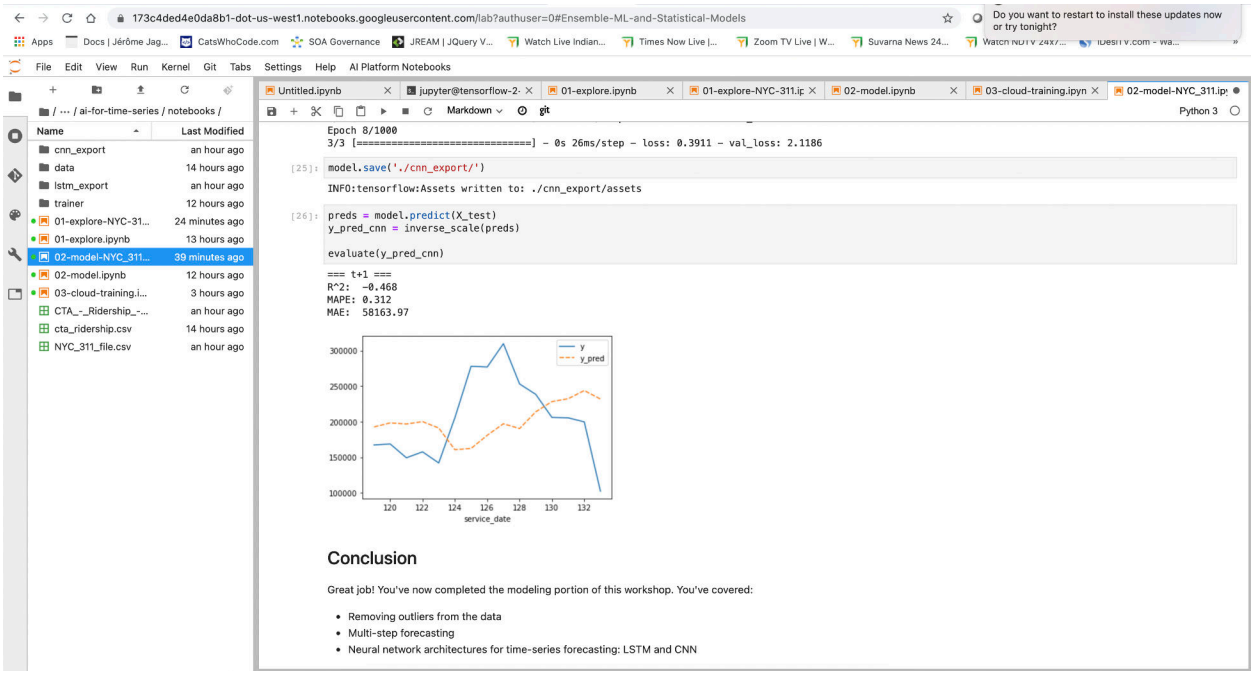

173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0#Ensemble-ML-and-Statistical-Models
Do you want to restart to install these updates now or try tonight?
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
Untitled.ipynb
jupyter@tensorflow-2·
01-explore.ipynb
01-explore-NYC-311.ip
02-model.ipynb
03-cloud-training.ipyn
02-model-NYC_311.ip
Markdown
git
Python 3
/ ··· / ai-for-time-series / notebooks /
Name
Last Modified
cnn_export an hour ago
data 14 hours ago
lstm_export an hour ago
trainer 12 hours ago
01-explore-NYC-31... 23 minutes ago
01-explore.ipynb 13 hours ago
02-model-NYC_311... 38 minutes ago
02-model.ipynb 12 hours ago
03-cloud-training.i... 3 hours ago
CTA_-_Ridership_-... an hour ago
cta_ridership.csv 13 hours ago
NYC_311_file.csv an hour ago
Scale the inputs and outputs
[17]:
For neural networks to converge quicker, it is helpful to scale the values.
For example, each feature might be transformed to have a mean of 0 and std. dev. of 1.
#
# You are working with a mix of features, input timesteps, output horizon, etc.
# which don't work out-of-the-box with common scaling utilities.
# So, here are a couple wrappers to handle scaling and inverting the scaling.
feature_scaler = StandardScaler()
target_scaler = StandardScaler()
def scale(df,
fit=True,
target_col=target_col,
feature_scaler=feature_scaler,
target_scaler=target_scaler):
"""
Scale the input features, using a separate scaler for the target.
Parameters:
df (pd.DataFrame): Input dataframe
fit (bool): Whether to fit the scaler to the data (only apply to training data)
target_col (pd.Series): The column that is being predicted
feature_scaler (StandardScaler): Scaler used for features
target_scaler (StandardScaler): Scaler used for target
Returns:
df_scaled (pd.DataFrame): Scaled dataframe
"""
target = df[target_col].values.reshape(-1, 1)
if fit:
target_scaler.fit(target)
target_scaled = target_scaler.transform(target)
# Select all columns other than target to be features
features = df.loc[:, df.columns != target_col].values
if features.shape[1]: # If there are any features
if fit:
feature_scaler.fit(features)

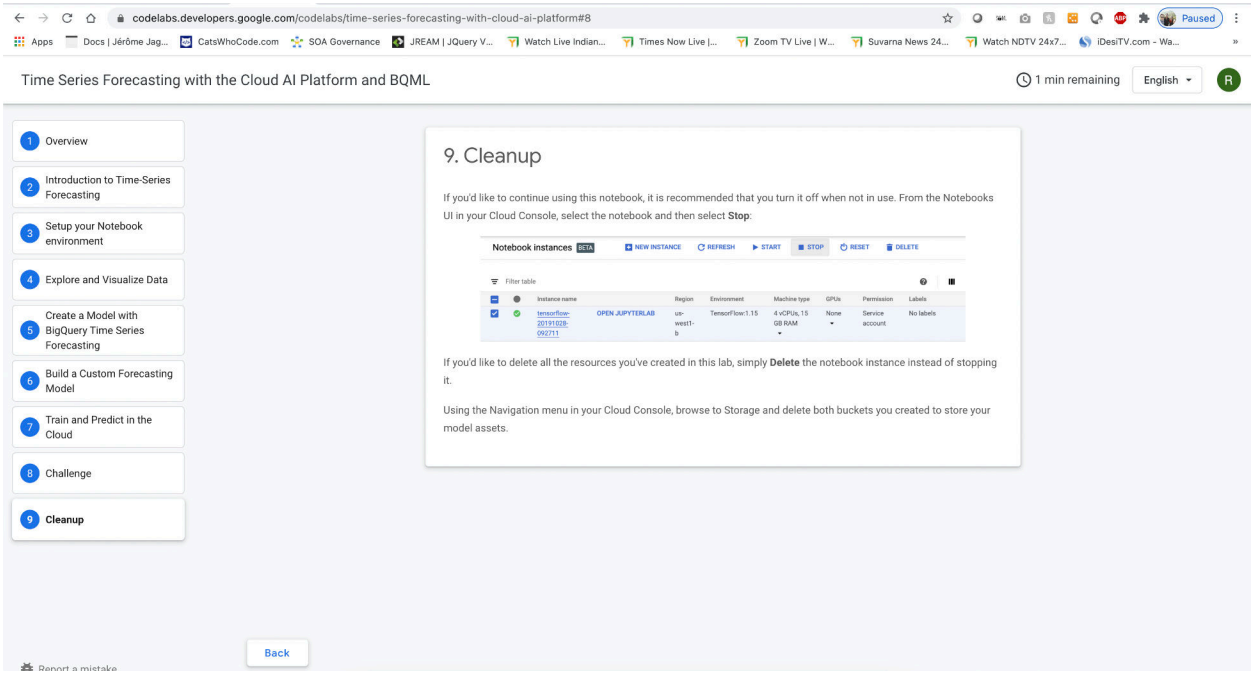

173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0#Ensemble-ML-and-Statistical-Models
Do you want to restart to install these updates now or try tonight?
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
/ ··· / ai-for-time-series / notebooks /
Name
Last Modified
cnn_export an hour ago
data 14 hours ago
lstm_export an hour ago
trainer 12 hours ago
01-explore-NYC-31... 24 minutes ago
01-explore.ipynb 13 hours ago
02-model-NYC_311... 39 minutes ago
02-model.ipynb 12 hours ago
03-cloud-training.i... 3 hours ago
CTA_-_Ridership_-... an hour ago
cta_ridership.csv 13 hours ago
NYC_311_file.csv an hour ago
Untitled.ipynb
jupyter@tensorflow-2·
01-explore.ipynb
01-explore-NYC-311.ip
02-model.ipynb
03-cloud-training.ipyn
02-model-NYC_311.ip
Markdown
git
Python 3
Create sequences of time-series data
[18]:
def reframe(data, n_input_steps = n_input_steps, n_output_steps = n_output_steps, target_col = target_col):
target_col_num = data.columns.get_loc(target_col)
# Iterate through data and create sequences of features and outputs
df = pd.DataFrame(data)
cols=list()
for i in range(n_input_steps, 0, -1):
cols.append(df.shift(i))
for i in range(0, n_output_steps):
cols.append(df.shift(-i))
# Concatenate values and remove any missing values
df = pd.concat(cols, axis=1)
df.dropna(inplace=True)
# Split the data into feature and target variables
n_feature_cols = n_input_steps * n_features
features = df.iloc[:,0:n_feature_cols]
target_cols = [i for i in range(n_feature_cols + target_col_num, n_feature_cols + n_output_steps * n_features, n_features)]
targets = df.iloc[:,target_cols]
return (features, targets)
X_train_reframed, y_train_reframed = reframe(df_train_scaled)
X_test_reframed, y_test_reframed = reframe(df_test_scaled)
Evaluate results
[19]:
def print_stats(timestep, y_true, y_pred, target_col, chart=True, table=False, dec=3):
'''
Helper function to print overall summary statistics and stats for each time step
'''
# Print summary statistics
print('=== t+' + str(timestep) + ' ===')
print('R^2: ' + str(np.round(r2_score(y_true, y_pred), dec)))
print('MAPE: ' + str(np.round(mean_absolute_percentage_error(y_true, y_pred), dec)))

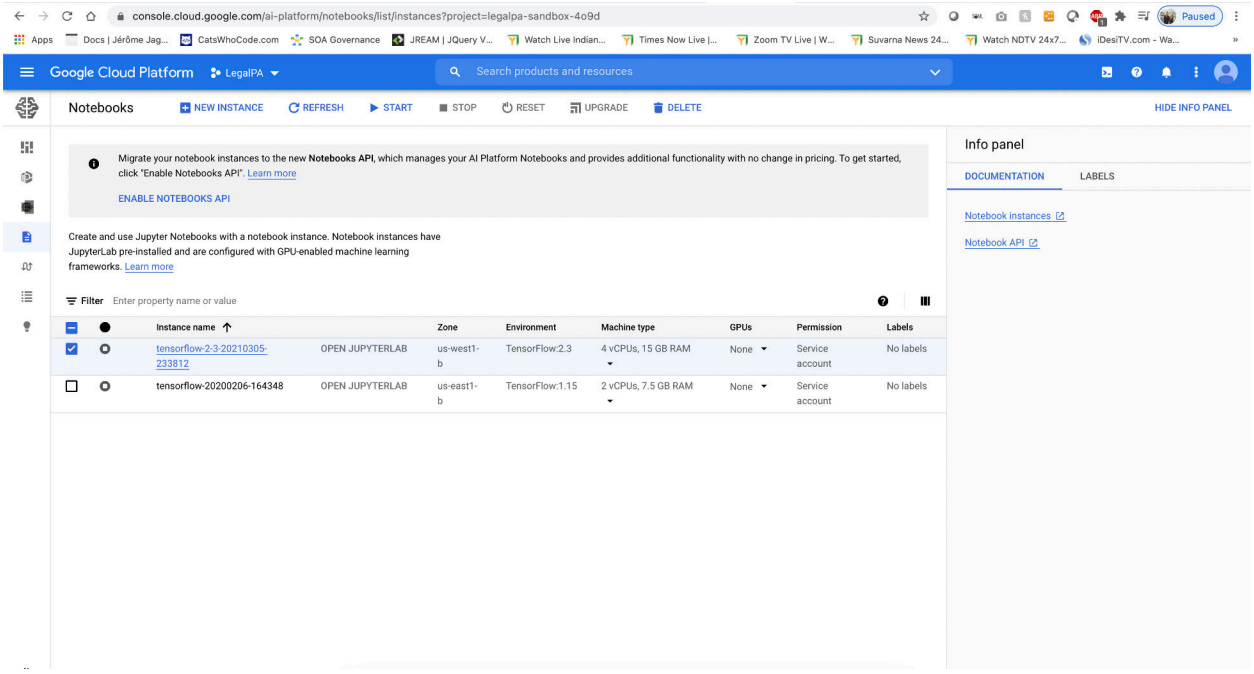

173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0#Ensemble-ML-and-Statistical-Models
Do you want to restart to install these updates now or try tonight?
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
/ ··· / ai-for-time-series / notebooks /
Name
Last Modified
cnn_export an hour ago
data 14 hours ago
lstm_export an hour ago
trainer 12 hours ago
01-explore-NYC-31... 24 minutes ago
01-explore.ipynb 13 hours ago
02-model-NYC_311... 39 minutes ago
02-model.ipynb 12 hours ago
03-cloud-training.i... 3 hours ago
CTA_-_Ridership_-... an hour ago
cta_ridership.csv 13 hours ago
NYC_311_file.csv an hour ago
Untitled.ipynb
jupyter@tensorflow-2·
01-explore.ipynb
01-explore-NYC-311.ip
02-model.ipynb
03-cloud-training.ipyn
02-model-NYC_311.ip
Markdown
git
Python 3
Long Short Term Memory (LSTM)
[20]:
# Reshape test data to match model inputs and outputs
X_train = X_train_reframed.values.reshape(-1, n_input_steps, n_features)
X_test = X_test_reframed.values.reshape(-1, n_input_steps, n_features)
y_train = y_train_reframed.values.reshape(-1, n_output_steps, 1)
y_test = y_test_reframed.values.reshape(-1, n_output_steps, 1)
TODO 2: Update the LSTM architecture
Try increasing and decreasing the number of LSTM units and see if you notice any accuracy improvements.
You can use hyper-parameter tuning to search for optimal values, but that's outside the scope of this lab.
[21]:
# Try increasing and decreasing the number of LSTM units and see if you notice any accuracy improvements.
# Run the next cell to evaluate the results in more detail.
model = Sequential([
LSTM(64, input_shape=[n_input_steps, n_features]),
Dense(n_output_steps)])
model.compile(optimizer='adam', loss='mae')
early_stopping = EarlyStopping(monitor='val_loss', patience=patience)
_ = model.fit(x=X_train, y=y_train, validation_data=(X_test, y_test), epochs=epochs, callbacks=[early_stopping])
Epoch 1/1000
3/3 [==========] - 1s 333ms/step - loss: 0.9243 - val_loss: 1.8443
Epoch 2/1000
3/3 [==========] - 0s 39ms/step - loss: 0.8203 - val_loss: 1.8219
Epoch 3/1000
3/3 [==========] - 0s 40ms/step - loss: 0.7305 - val_loss: 1.8052
Epoch 4/1000
3/3 [==========] - 0s 42ms/step - loss: 0.6456 - val_loss: 1.7969
Epoch 5/1000
3/3 [==========] - 0s 45ms/step - loss: 0.5665 - val_loss: 1.8299
Epoch 6/1000
3/3 [==========] - 0s 41ms/step - loss: 0.4644 - val_loss: 1.9126
Epoch 7/1000
3/3 [==========] - 0s 38ms/step - loss: 0.4168 - val_loss: 2.0431


173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0#Ensemble-ML-and-Statistical-Models
Do you want to restart to install these updates now or try tonight?
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
/ ··· / ai-for-time-series / notebooks /
Name Last Modified
cnn_export an hour ago
data 14 hours ago
lstm_export an hour ago
trainer 12 hours ago
01-explore-NYC-31... 24 minutes ago
01-explore.ipynb 13 hours ago
02-model-NYC_311... 39 minutes ago
02-model.ipynb 12 hours ago
03-cloud-training.i... 3 hours ago
CTA_-_Ridership_-... an hour ago
cta_ridership.csv 13 hours ago
NYC_311_file.csv an hour ago
Untitled.ipynb
jupyter@tensorflow-2·
01-explore.ipynb
01-explore-NYC-311.ip
02-model.ipynb
03-cloud-training.ipyn
02-model-NYC_311.ip
Markdown
git
Python 3
[23]:
# Predict the results, and then reverse the transformation that scaled all values to a mean of 0 and std. dev. of 1
preds = model.predict(X_test)
y_pred_lstm = inverse_scale(preds)
# Evaluate the overall results and for each time step
evaluate(y_pred_lstm)
# The plot will show the R^2 value (0 lowest -> 1 highest) and the MAE (mean absolute error) for the entire prediction window.
# It will also show individual plots for 1 day out, 2 days out, etc. comparing the actual vs the predicted value.
=== t+1 ===
R^2: -0.578
MAPE: 0.324
MAE: 59756.94
y
y_pred
300000
250000
200000
150000
100000
120
122
124
126
128
130
132
service_date
Convolutional Neural Network (CNN)
TODO 3: Update the CNN architecture
Try adjusting the # of filters (pattern types) and kernel size (size of the sliding window)
[24]:
from tensorflow.keras.layers import AveragePooling1D
# TODO: Try adjusting the # of filters (pattern types) and kernel size (size of the sliding window)
model = Sequential([


173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0#Ensemble-ML-and-Statistical-Models
Do you want to restart to install these updates now or try tonight?
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
/ ··· / ai-for-time-series / notebooks /
Name Last Modified
cnn_export an hour ago
data 14 hours ago
lstm_export an hour ago
trainer 12 hours ago
01-explore-NYC-31... 24 minutes ago
01-explore.ipynb 13 hours ago
02-model-NYC_311... 39 minutes ago
02-model.ipynb 12 hours ago
03-cloud-training.i... 3 hours ago
CTA_-_Ridership_-... an hour ago
cta_ridership.csv 13 hours ago
NYC_311_file.csv an hour ago
Untitled.ipynb
jupyter@tensorflow-2·
01-explore.ipynb
01-explore-NYC-311.ip
02-model.ipynb
03-cloud-training.ipyn
02-model-NYC_311.ip
Markdown
git
Python 3
Convolutional Neural Network (CNN)
TODO 3: Update the CNN architecture
Try adjusting the # of filters (pattern types) and kernel size (size of the sliding window)
[24]:
from tensorflow.keras.layers import AveragePooling1D
# TODO: Try adjusting the # of filters (pattern types) and kernel size (size of the sliding window)
model = Sequential([
Conv1D(filters=32, kernel_size=3, input_shape=[n_input_steps, n_features]),
Flatten(),
Dense(n_output_steps)])
model.compile(optimizer='adam', loss='mae')
early_stopping = EarlyStopping(monitor='val_loss', patience=5)
_ = model.fit(x=X_train, y=y_train, validation_data=(X_test, y_test), epochs=epochs, callbacks=[early_stopping])
Epoch 1/1000
3/3 [==========] - 0s 99ms/step - loss: 0.9515 - val_loss: 1.8508
Epoch 2/1000
3/3 [==========] - 0s 31ms/step - loss: 0.8268 - val_loss: 1.8233
Epoch 3/1000
3/3 [==========] - 0s 27ms/step - loss: 0.7060 - val_loss: 1.8182
Epoch 4/1000
3/3 [==========] - 0s 27ms/step - loss: 0.5980 - val_loss: 1.8426
Epoch 5/1000
3/3 [==========] - 0s 29ms/step - loss: 0.5055 - val_loss: 1.8882
Epoch 6/1000
3/3 [==========] - 0s 26ms/step - loss: 0.4340 - val_loss: 1.9874
Epoch 7/1000
3/3 [==========] - 0s 26ms/step - loss: 0.3882 - val_loss: 2.0666
Epoch 8/1000
3/3 [==========] - 0s 26ms/step - loss: 0.3911 - val_loss: 2.1186
[25]:
model.save('./cnn_export/')
INFO:tensorflow:Assets written to: ./cnn_export/assets
[26]:
preds = model.predict(X_test)
y_pred_cnn = inverse_scale(preds)


173c4ded4e0da8b1-dot-us-west1.notebooks.googleusercontent.com/lab?authuser=0#Ensemble-ML-and-Statistical-Models
Do you want to restart to install these updates now or try tonight?
File Edit View Run Kernel Git Tabs Settings Help AI Platform Notebooks
Untitled.ipynb
jupyter@tensorflow-2-
01-explore.ipynb
01-explore-NYC-311.ip
02-model.ipynb
03-cloud-training.ipyn
02-model-NYC_311.ip
Markdown
git
Python 3
/ ... / ai-for-time-series / notebooks /
Name
Last Modified
cnn_export an hour ago
data 14 hours ago
lstm_export an hour ago
trainer 12 hours ago
01-explore-NYC-31... 24 minutes ago
01-explore.ipynb 13 hours ago
02-model-NYC_311... 39 minutes ago
02-model.ipynb 12 hours ago
03-cloud-training.i... 3 hours ago
CTA_-_Ridership_-... an hour ago
cta_ridership.csv 14 hours ago
NYC_311_file.csv an hour ago
Epoch 8/1000
3/3 [==========] - 0s 26ms/step - loss: 0.3911 - val_loss: 2.1186
[25]: model.save('./cnn_export/')
INFO:tensorflow:Assets written to: ./cnn_export/assets
[26]: preds = model.predict(X_test)
y_pred_cnn = inverse_scale(preds)
evaluate(y_pred_cnn)
=== t+1 ===
R^2: -0.468
MAPE: 0.312
MAE: 58163.97
y
y_pred
300000
250000
200000
150000
100000
120 122 124 126 128 130 132
service_date
Conclusion
Great job! You've now completed the modeling portion of this workshop. You've covered:
Removing outliers from the data
Multi-step forecasting
Neural network architectures for time-series forecasting: LSTM and CNN


codelabs.developers.google.com/codelabs/time-series-forecasting-with-cloud-ai-platform#8
Paused
Time Series Forecasting with the Cloud AI Platform and BQML
1 min remaining
English
1 Overview
2 Introduction to Time-Series Forecasting
3 Setup your Notebook environment
4 Explore and Visualize Data
5 Create a Model with BigQuery Time Series Forecasting
6 Build a Custom Forecasting Model
7 Train and Predict in the Cloud
8 Challenge
9 Cleanup
9. Cleanup
If you'd like to continue using this notebook, it is recommended that you turn it off when not in use. From the Notebooks UI in your Cloud Console, select the notebook and then select Stop:
Notebook instances BETA
NEW INSTANCE
REFRESH
START
STOP
RESET
DELETE
Filter table
Instance name
Region
Environment
Machine type
GPUs
Permission
Labels
tensorflow-20191028-092711
OPEN JUPYTERLAB
us-west1-b
TensorFlow:1.15
4 vCPUs, 15 GB RAM
None
Service account
No labels
If you'd like to delete all the resources you've created in this lab, simply Delete the notebook instance instead of stopping it.
Using the Navigation menu in your Cloud Console, browse to Storage and delete both buckets you created to store your model assets.
Back
Report a mistake


console.cloud.google.com/ai-platform/notebooks/list/instances?project=legalpa-sandbox-4o9d
Paused
Apps
Docs | Jérôme Jag...
CatsWhoCode.com
SOA Governance
JREAM | JQuery V...
Watch Live Indian...
Times Now Live |...
Zoom TV Live | W...
Suvarna News 24...
Watch NDTV 24x7...
iDesiTV.com - Wa...
Google Cloud Platform
LegalPA
Search products and resources
Notebooks
NEW INSTANCE
REFRESH
START
STOP
RESET
UPGRADE
DELETE
HIDE INFO PANEL
Migrate your notebook instances to the new Notebooks API, which manages your AI Platform Notebooks and provides additional functionality with no change in pricing. To get started, click "Enable Notebooks API". Learn more
ENABLE NOTEBOOKS API
Create and use Jupyter Notebooks with a notebook instance. Notebook instances have JupyterLab pre-installed and are configured with GPU-enabled machine learning frameworks. Learn more
Filter
Enter property name or value
Instance name
Zone
Environment
Machine type
GPUs
Permission
Labels
tensorflow-2-3-20210305-233812
OPEN JUPYTERLAB
us-west1-b
TensorFlow:2.3
4 vCPUs, 15 GB RAM
None
Service account
No labels
tensorflow-20200206-164348
OPEN JUPYTERLAB
us-east1-b
TensorFlow:1.15
2 vCPUs, 7.5 GB RAM
None
Service account
No labels
Info panel
DOCUMENTATION
LABELS
Notebook instances
Notebook API