```
# IMPORTANT: RUN THIS CELL IN ORDER TO IMPORT YOUR KAGGLE DATA SOUR
# TO THE CORRECT LOCATION (/kaggle/input) IN YOUR NOTEBOOK,
# THEN FEEL FREE TO DELETE THIS CELL.
# NOTE: THIS NOTEBOOK ENVIRONMENT DIFFERS FROM KAGGLE'S PYTHON
# ENVIRONMENT SO THERE MAY BE MISSING LIBRARIES USED BY YOUR
# NOTEBOOK.

import os
import sys
from tempfile import NamedTemporaryFile
from urllib.request import urlopen
from urllib.parse import unquote, urlparse
from urllib.error import HTTPError
from zipfile import ZipFile
import tarfile
import shutil

CHUNK_SIZE = 40960
DATA_SOURCE_MAPPING = 'audio-speech-sentiment:https%3A%2F%2Fstorage

KAGGLE_INPUT_PATH='/kaggle/input'
KAGGLE_WORKING_PATH='/kaggle/working'
KAGGLE_SYMLINK='kaggle'

!umount /kaggle/input/ 2> /dev/null
shutil.rmtree('/kaggle/input', ignore_errors=True)
os.makedirs(KAGGLE_INPUT_PATH, 0o777, exist_ok=True)
os.makedirs(KAGGLE_WORKING_PATH, 0o777, exist_ok=True)

try:
  os.symlink(KAGGLE_INPUT_PATH, os.path.join("..", 'input'), target_
except FileExistsError:
  pass
try:
  os.symlink(KAGGLE_WORKING_PATH, os.path.join("..", 'working'), ta
except FileExistsError:
  pass

for data_source_mapping in DATA_SOURCE_MAPPING.split(','):
    directory, download_url_encoded = data_source_mapping.split(':'
    download_url = unquote(download_url_encoded)
    filename = urlparse(download_url).path
    destination_path = os.path.join(KAGGLE_INPUT_PATH, directory)
    try:
        with urlopen(download_url) as fileres, NamedTemporaryFile()
            total_length = fileres.headers['content-length']
            print(f'Downloading {directory}, {total_length} bytes c
            dl = 0
            data = fileres.read(CHUNK_SIZE)
            while len(data) > 0:
                dl += len(data)
                tfile.write(data)
                done = int(50 * dl / int(total_length))
                sys.stdout.write(f"\r[{'=' * done}{' ' * (50-done)}
                sys.stdout.flush()
                data = fileres.read(CHUNK_SIZE)
            if filename.endswith('.zip'):
              with ZipFile(tfile) as zfile:
                zfile.extractall(destination_path)
            else:
              with tarfile.open(tfile.name) as tarfile:
                tarfile.extractall(destination_path)
            print(f'\nDownloaded and uncompressed: {directory}')
    except HTTPError as e:
```

Release notes

Please follow our blog to see more information about new features, tips and tricks, and featured notebooks such as Analyzing a Bank Failure with Colab.

2024-11-11

- Users can now import Gemini API keys from AI Studio into their user secrets, all in Colab (tweet).
- Increased limit to 1000 characters for requests to Gemini in Chat and Generate windows.
- Improved saving notebook to GitHub flow.
- Updated Gemini spark icon to be colorful
- uv is pre-installed on the PATH for faster package installs.
- Fixed bugs
    - Dropdown text for GitHub repository not visible #4901.
    - Pre-installed California housing dataset README not correct #4862.
    - Backend execution error for scheduled notebook #4850.
    - Drive File Stream issues #3441.
    - Linking to the signup page does not preserve the authuser parameter.
    - Error messages in Gemini chat are not polished.
    - Clicking in Gemini chat feedback causes jitters the UI.
    - Hovering over a table of contents entry would show the menu icons for all entries.
    - Surveys display over open dialogs.
    - Playground mode banner not shown on mobile.

Python package upgrades

- accelerate 0.34.2 -> 1.1.1
- arviz 0.19.0 -> 0.20.0
- bigframes 1.18.0 -> 1.25.0
- bigquery-magics 0.2.0 -> 0.4.0
- bokeh 3.4.3 -> 3.6.1
- blosc 2.0.0 -> 2.7.1
- cloudpickle 2.2.1 -> 3.1.0
- cudf-cu12 24.4.1 -> 24.10.1
- dask 2024.8.0 -> 24.10.0
- debugpy 1.6.6 -> 1.8.0
- earthengine-api 1.0.0 -> 1.2.0
- folium 0.17.0 -> 0.18.0
- gscfs 2024.6.1 -> 2024.10.0
- geemap 0.34.3 -> 0.35.1
- holidays 0.57 -> 0.60
- huggingface-hub 0.24.7 -> 0.26.2
- kagglehub 0.3.0 -> 0.3.3
- lightgbm 4.4.0 -> 4.5.0
- lxml 4.9.4 -> 5.3.0
- matplotlib 3.7.1 -> 3.8.0
- mizani 0.11.4 -> 0.13.0
- networkx 3.3 -> 3.4.2
- nltk 3.8.1 -> 3.9.1
- pandas 2.1.4 -> 2.2.2
- pillow 10.4.0 -> 11.0.0
- plotnine 0.13.6 -> 0.14.1
- polars 1.6.0 -> 1.9.0

```
    except HTTPError as e:
        print(f'Failed to load (likely expired) {download_url} to p
        continue
    except OSError as e:
        print(f'Failed to load {download_url} to path {destination_
        continue

print('Data source import complete.')
```

```
Downloading audio-speech-sentiment, 242603488 bytes compressec
[==================================================] 24260348
Downloaded and uncompressed: audio-speech-sentiment
Data source import complete.
```

Audio Sentiment Analysis

The aim of this challenge is to read the audio (.wav) files and classify them into 3 sentiments (Positive, Neutral, or Negative).

Sentiments:-

- Positive
- Negative
- Neutral

We will be applying following Ensemble Algorithms:-

- NN with Tensorflow

Reading & Understanding Data

Importing Libraries

```
import sys, os
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
sns.set_style('whitegrid')
%matplotlib inline
import warnings
warnings.filterwarnings('ignore')
import sklearn.metrics as skm
import sklearn.model_selection as skms
import sklearn.preprocessing as skp
import random, os
import librosa, IPython
import librosa.display as lplt
from skimage.io import imread
seed = 12
np.random.seed(seed)
```

Loading Dataset

```
trainPath = '/kaggle/input/audio-speech-sentiment/TRAIN/'
testPath = '/kaggle/input/audio-speech-sentiment/TEST/'
```

- protobuf 3.20.3 -> 4.25.5
- pyarrow 14.0.2 -> 17.0.0
- pydrive2 1.20.0 -> 1.21.1
- pymc 5.16.2 -> 5.18.0
- torch 2.4.1 -> 2.5.0
- torchaudio 2.4.1 -> 2.5.0
- torchvision 0.19.1 -> 0.20.0
- transformers 4.44.2 -> 4.46.2
- xarray 2024.9.0 -> 2024.10.0

Python package inclusions

- diffusers 0.31.0
- gitpython 3.1.43
- langchain 0.3.7
- openai 1.54.3
- pygit2 1.16.0
- pyspark 3.5.3
- sentence-transformers 3.2.1
- timm 1.0.11
- wandb 0.18.6

Library and driver upgrades

- drivefs upgraded from 89.0.2 to 98.0.0

2024-09-23

- Improved code snippet search
- Updated Marketplace image and public local runtime container
- Improved the look-and-feel of interactive form dropdowns and checkboxes
- Fixed bugs
    - activating the skip link caused the notebook to scroll out of view
    - toggling a checkbox too much caused the page to crash
    - lightning fast drags could cause orphaned tabs
    - custom widgets snippet would show for local runtimes

Python package upgrades

- accelerate 0.32.1 -> 0.34.2
- arviz 0.18.0 -> 0.19
- autograd 1.6.2 -> 1.7.0
- bigframes 1.14.0 -> 1.18.0
- dask 2024.7.1 -> 2024.8.0
- distributed 2024.7.1 -> 2024.8.0
- duckdb 0.10.3 -> 1.1.0
- earthengine-api 0.1.416 -> 1.0.0
- flax 0.8.4 -> 0.8.5
- gdown 5.1.0 -> 5.2.0
- geemap 0.33.1 -> 0.34.3
- geopandas 0.14.4 -> 1.0.1
- google-cloud-aiplatform 1.59.0 -> 1.67.1
- google-cloud-bigquery-storage 2.25.0 -> 2.26.0
- holidays 0.54 -> 0.57
- huggingface-hub 0.23.5 -> 0.24.7
- ibis-framework 8.0.0 -> 9.2.0
- jax 0.4.26 -> 0.4.33
- jaxlib 0.4.26 -> 0.4.33
- kagglehub 0.2.9 -> 0.3.0
- lightgbm 4.4.0 -> 4.5.0
- matplotlib-venn 0.11.10 -> 1.1.1
- mizani 0.9.3 -> 0.11.4
- Pillow 9.4.0 -> 10.4.0
- plotly 5.15.0 -> 5.24.1
- plotnine 0.12.4 -> 0.13.6
- polars 0.20.2 -> 1.6.0
- progressbar2 4.2.0 -> 4.5.0
- PyDrive2 1.6.3 -> 1.20.0
- pymc 5.10.4 -> 5.16.2

```
df_base = pd.read_csv('/kaggle/input/audio-speech-sentiment/TRAIN.
df_base.head()
```

	Filename	Class
0	346.wav	Negative
1	163.wav	Neutral
2	288.wav	Negative
3	279.wav	Negative
4	244.wav	Negative

About the dataset

```
print("Dataset has",df_base.shape[0],"samples")
print("Count of Positive and Negative samples")
df_base['Class'].value_counts().reset_index()
```

```
Dataset has 250 samples
Count of Positive and Negative samples
```

	Class	count
0	Negative	87
1	Positive	82
2	Neutral	81

```
sample_rate = 44100
def loadAudio(fp):
    return librosa.load(fp, res_type='kaiser_fast', duration=2.5,
```

MelSpec -> Array

```
def scanFeatures(path, avgFeat=0):
    features = []
    minFeat = sys.maxsize
    maxFeat = 0
    files = sorted(os.listdir(path))
    print("Scanning", path)

    for i, fp in enumerate(files):
        X, sr = loadAudio(os.path.join(path, fp))

        f = librosa.feature.melspectrogram(y=X, sr=sample_rate)
        f = librosa.amplitude_to_db(f, ref=np.max)

        shapeY = f.shape[1]
        if shapeY < minFeat:
            minFeat = shapeY

        if shapeY > maxFeat:
            maxFeat = shapeY

        features.append(f)
    if avgFeat == 0:
        avgFeat = int((minFeat+maxFeat)/2)
    feat_mat = np.zeros((len(files), f.shape[0], avgFeat))
    for i, x in enumerate(features):
```

- pytensor 2.18.6 -> 2.25.4
- scikit-image 0.23.2 -> 0.24.0
- scikit-learn 1.3.2 -> 1.5.2
- torch 2.3.1 -> 2.4.1
- torchaudio 2.3.1 -> 2.4.1
- torchvision 0.18.1 -> 0.19.1
- transformers 4.42.4 -> 4.44.2
- urllib3 2.0.7 -> 2.2.3
- xarray 2024.6.0 -> 2024.9.0

Python package inclusions

- bigquery-magics 0.2.0

2024-08-20

- TPU memory usage and utilization can now be checked with `!tpu-info`
- Gemini Chat responses are now grounded in relevant sources
- Added a new "Create Gemini API key" link in the user secrets panel
- Added a new "Gemini: Creating a prompt" snippet and touched up the existing "Gemini: Connecting to Gemini" snippet
- Added the ability to specify custom placeholder text for various interactive form params (see examples)
- Keyboard navigation a11y improvements to comments UI
- Various minor rendering improvements to interactive forms UI
- A11y improvements for the run button and header
- Updated tooltip styling
- A11y improvements for the file browser's disk usage bar
- On mobile, tooltips now trigger on long press
- On mobile, release notes updates will no longer display automatically
- Python package upgrades
    - astropy 5.3.4 -> 6.1.2
    - bigframes 1.11.1 -> 1.14.0
    - bokeh 3.3.4 -> 3.4.3
    - dask 2023.8.1 -> 2024.7.1
    - earthengine-api 0.1.412 -> 0.1.416
    - geopandas 0.13.2 -> 0.14.4
    - kagglehub 0.2.8 -> 0.2.9
    - keras 2.15.0 -> 3.4.1
    - lightgbm 4.1.0 -> 4.4.0
    - malloy 2023.1067 -> 2024.1067
    - numba 0.58.1 -> 0.60.0
    - numpy 1.25.2 -> 1.26.4
    - opencv-python 4.8.0.76 -> 4.10.0.84
    - pandas 2.0.3 -> 2.1.4
    - pandas-gbq 0.19.2 -> 0.23.1
    - panel 1.3.8 -> 1.4.5
    - requests 2.31.0 -> 2.32.3
    - scikit-learn 1.2.2. -> 1.3.2
    - scipy 1.11.4 -> 1.13.1
    - tensorboard 2.15.2 -> 2.17.0
    - tensorflow 2.15.0 -> 2.17.0
    - tf-keras 2.15.1 -> 2.17.0
    - xarray 2023.7.0 -> 2024.6.0
    - xgboost 2.0.3 -> 2.1.1
- Python package inclusions

```
        xWidth = min(x.shape[1],avgFeat)
        feat_mat[i, :, :xWidth] = x[:,:xWidth]
    return feat_mat, files


f_dim = 128
train_data, train_files = scanFeatures(trainPath, f_dim)
test_data, test_files = scanFeatures(testPath, train_data.shape[1]
print(train_data.shape)
print(test_data.shape)
```

```
Scanning /kaggle/input/audio-speech-sentiment/TRAIN/
Scanning /kaggle/input/audio-speech-sentiment/TEST/
(250, 128, 128)
(110, 128, 128)
```

MelSpec -> Images

```
def saveImg(f, fp):
    f = np.flip(f, axis=0)
    plt.figure()
    plt.axis('off')
    plt.imsave(fp, f, format='png')
    plt.clf()


def saveFeatureToImage(path, saveDir, avgFeat=0):
    global sample_rate
    files = sorted(os.listdir(path))
    print("Scanning", path)

    for i, fp in enumerate(files):
        X, sr = loadAudio(os.path.join(path, fp))

        f = librosa.feature.melspectrogram(y=X, sr=sample_rate)
        f = librosa.amplitude_to_db(f, ref=np.max)

        img = np.zeros((f.shape[0], avgFeat))
        xWidth = min(f.shape[1],avgFeat)
        img[:, :xWidth] = f[:,:xWidth]
        fname = os.path.join(saveDir, fp.split('.')[0] + '.png')
        saveImg(img, fname)


f_dim = 128
train_img_dir = './train_images'
test_img_dir = './test_images'
if not os.path.exists(train_img_dir):
    os.mkdir(train_img_dir)
    saveFeatureToImage(trainPath, train_img_dir, f_dim)
if not os.path.exists(test_img_dir):
    os.mkdir(test_img_dir)
    saveFeatureToImage(testPath, test_img_dir, train_data.shape[1]
```

- einops 0.8.0

2024-07-22

- You can now embed Google sheets directly into Colab to streamline interactions with data with InteractiveSheet.

  Example:

  ```
  from google.colab import sheets
  sh = sheets.InteractiveSheet()
  df = sh.as_df()
  ```
- Fixed multiple rendering bugs in cell editors with wide text content (i.e. text is no longer hidden or clipped)
- Fixed multiple accessibility issues in Colab's comments feature (e.g. proper keyboard focus management, added accessibility landmarks, etc)
- Fixed bug where AI code generation would fail for extremely long broken code snippets
- Fixed multiple scrollbar bugs in the user secrets panel
- Added the ability for workspace admin to purchase Colab Pro and Pro+ Subscriptions for users
- Fixed bug where user secrets couldn't be moved to a tab
- Fixed several focus management accessibility issues in tabs, the table of contents, the left toolbar, and the run button
- Fixed bug where overflowing cells may be omitted when pasting from Google Sheets
- Fixed bug where the generate code button did not activate on touch
- Python package upgrades
  - bigframes 1.9.0 -> 1.11.1
  - cvxpy 1.3.4 -> 1.5.2
  - earthengine-api 0.1.408 -> 0.1.412
  - google-api-core 2.11.1 -> 2.19.1
  - google-api-python-client 2.84.0 -> 2.137.0
  - google-cloud-aiplatform 1.56.0 -> 1.59.0
  - google-cloud-bigquery 3.21.0 -> 3.25.0
  - google-cloud-core 2.3.3 -> 2.4.1
  - google-cloud-datastore 2.15.2 -> 2.19.0
  - google-cloud-firestore 2.11.1 -> 2.16.1
  - google-cloud-functions 1.13.3 -> 1.16.4
  - google-generativeai 0.5.4 -> 0.7.2
  - kagglehub 0.2.5 -> 0.2.8
  - pip 23.1.2 -> 24.1.2
  - setuptools 67.7.2 -> 71.0.4
  - sympy 1.12.1 -> 1.13.1
  - torch 2.3.0 -> 2.3.1
  - transformers 4.41.2 -> 4.42.4
- Python package inclusions
  - accelerate 0.32.1

2024-06-18

```
Scanning /kaggle/input/audio-speech-sentiment/TRAIN/
Scanning /kaggle/input/audio-speech-sentiment/TEST/
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
<Figure size 640x480 with 0 Axes>
```

```
def scanImgFeatures(path):
    features = []
    files = sorted(os.listdir(path))
    for x in files:
        fp = os.path.join(path, x)
        img = imread(fp)[:,:,:3]/255.0
        features.append(img)
    return np.array(features), files
```

```
<Figure size 640x480 with 0 Axes>
```

```
if os.path.exists(train_img_dir):
    train_data_img, train_files_img = scanImgFeatures(train_img_di
if os.path.exists(test_img_dir):
```

- Inline AI completions are now available to users on the free-of-charge tier
- Reduced latency for LSP and terminal connections
- Improved quality of inline completions
- Visual improvements to switch controls across Colab
- Various bug fixes, performance and a11y improvements to the user secrets panel
- Improved tooltip UX behavior
- Improved behavior when copying data from Google Sheets and pasting in Colab
- Scroll to cell fixes for single tabbed view and jump to cell command
- Improved tab header behavior
- A11y improvements for notebook-focused cells
- Python package upgrades
  - torch 2.2.1 -> 2.3.0
  - torchaudio 2.2.1 -> 2.3.0
  - torchvision 0.17.1 -> 0.18.0
  - torchtext 0.17.1 -> 0.18.0
  - google-cloud-aiplatform 1.51.0 -> 1.56.0
  - bigframes 1.5.0 -> 1.8.0
  - regex 2023.12.25 -> 2024.5.15

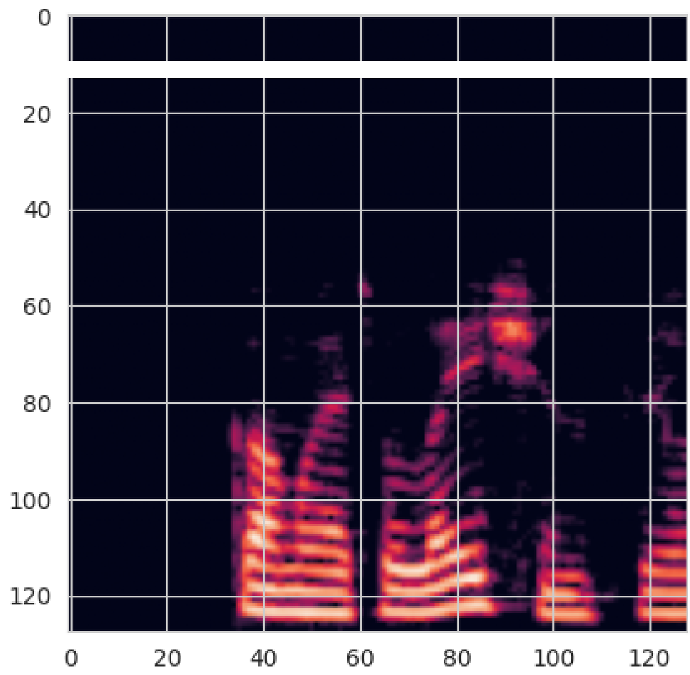
2024-05-13

- Code actions are now supported to automatically improve and refactor code. Code actions can be triggered by the keyboard shortcut "Ctrl/⌘ + ."
- Python package upgrades
  - bigframes 1.0.0 -> 1.5.0
  - google-cloud-aiplatform 1.47.0 -> 1.51.0
  - jax[tpu] 0.4.23 -> 0.4.26
- Python package inclusions
  - cudf 24.4.1

2024-04-15

- TPU v2 runtime is now available
- L4 runtime is now available for paid users
- New distributed fine-tuning Gemma tutorial on TPUs (GitHub)
- Symbol rename is now supported with keyboard shortcut F2
- Fixed bug causing inability to re-upload deleted files
- Fixed breaking bug in colabtools %upload_files_async
- Added syntax highlighting to %%writefile cells
- Cuda dependencies that come with Torch are cached for faster downloads for packages that require Torch and its dependencies (GitHub issue)
- Python package upgrades
  - bigframes 0.24.0 -> 1.0.0
  - duckdb 0.9.2 -> 0.10.1
  - google-cloud-aiplatform 1.43.0 -> 1.47.0

```
    test_data_img, test_files_img = scanImgFeatures(test_img_dir)
    plt.imshow(test_data_img[0])
    plt.show()
```



```
     <Figure size 640x480 with 0 Axes>
```

```
def getPathLabels(p):
    return [df_base[df_base['Filename'] == x].iloc[0,1] for x in p
```

```
     <Figure size 640x480 with 0 Axes>
```

```
train_labels = getPathLabels(train_files)
```

```
     <Figure size 640x480 with 0 Axes>
```

Data Visualization

```
     <Figure size 640x480 with 0 Axes>
```

```
audio_fp = '/kaggle/input/audio-speech-sentiment/TRAIN/1.wav'
audio_data, sr = loadAudio(audio_fp)
audio_data, _ = librosa.effects.trim(audio_data)
```

```
     <Figure size 640x480 with 0 Axes>
     <Figure size 640x480 with 0 Axes>
```

```
# play sample file
IPython.display.Audio(audio_data, rate=sr)
```

0:00 / 0:01

```
     <Figure size 640x480 with 0 Axes>
```

```
# plot sample file
'''plt.figure(figsize=(15,5))
lplt.waveplot(audio_data)
plt.show()
```

- jax 0.4.23 -> 0.4.26

2024-03-13

- Fixed bug that sometimes caused UserSecrets to move / disappear
- Improved messaging for mounting drive in an unsupported environment (GitHub issue)
- Python package upgrades
  - torch 2.1.0 -> 2.2.1
  - torchaudio 2.1.0 -> 2.2.1
  - torchvision 0.16.0 -> 0.17.1
  - torchtext 0.16.0 -> 0.17.1
  - PyMC 5.7.2 -> 5.10.4
  - BigFrames 0.21.0 -> 0.24.0
  - google-cloud-aiplatform 1.42.1 -> 1.43.0
  - tornado 6.3.2 -> 6.3.3

2024-02-21

- Try out Gemma on Colab!
- Allow unicode in form text inputs
- Display documentation and link to source when displaying functions
- Display image-like ndarrays as images
- Improved UX around quick charts and execution error suggestions
- Released Marketplace image for the month of February (GitHub issue)
- Python package upgrades
  - bigframes 0.19.2 -> 0.21.0
  - regex 2023.6.3 -> 2023.12.25
  - spacy 3.6.1 -> 3.7.4
  - beautifulsoup4 4.11.2 -> 4.12.3
  - tensorflow-probability 0.22.0 -> 0.23.0
  - google-cloud-language 2.9.1 -> 2.13.1
  - google-cloud-aiplatform 1.39.0 -> 1.42.1
  - transformers 4.35.2 -> 4..37.2
  - pyarrow 10.0.1 -> 14.0.2

2024-01-29

- New Kaggle Notebooks <> Colab updates! Now you can:
  - Import directly from Colab without having to download/re-upload
  - Upload via link, by pasting Google Drive or Colab URLs
  - Export & run Kaggle Notebooks on Colab with 1 click
- Try these notebooks that talk to Gemini:
  - Gemini and Stable Diffusion
  - Learning with Gemini and ChatGPT
  - Talk to Gemini with Google's Speech to Text API
  - Sell lemonade with Gemini and Sheets
  - Generate images with Gemini and Vertex
- Python package upgrades
  - google-cloud-aiplatform 1.38.1 -> 1.39.0
  - bigframes 0.18.0 -> 0.19.2

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most
recent call last)
<ipython-input-19-963658b84229> in <cell line: 8>()
      6
      7 plt.figure(figsize=(15,5))
----> 8 librosa.display.waveplot(audio_data, sr=sr)
      9 plt.show()

AttributeError: module 'librosa.display' has no attribute
'waveplot'
<Figure size 640x480 with 0 Axes>
```

```
# Default FFT window size
n_fft = 2048 # window size
hop_length = 512 # window hop length for STFT

stft = librosa.stft(audio_data, n_fft=n_fft, hop_length=hop_lengtr
stft_db = librosa.amplitude_to_db(stft, ref=np.max)

plt.figure(figsize=(12,4))
lplt.specshow(stft, sr=sr, x_axis='time', y_axis='hz')
plt.colorbar()
plt.title("Spectrogram with amplitude")
plt.show()

plt.figure(figsize=(12,4))
lplt.specshow(stft_db, sr=sr, x_axis='time', y_axis='log', cmap='c
plt.colorbar()
plt.title("Spectrogram with decibel log")
plt.show()
```

  - polars 0.17.3 -> 0.20.2
  - gdown 4.6.6 -> 4.7.3 (GitHub issue)
  - tensorflow-hub 0.15.0 -> 0.16.0
  - flax 0.7.5 -> 0.8.0
- Python package inclusions
  - sentencepiece 0.1.99

2024-01-08

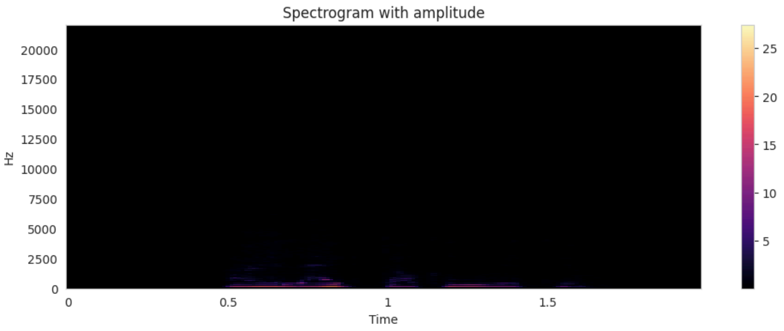
- Avoid nested scrollbars for large outputs by using `google.colab.output.no_vertical_s` Example notebook
- Fix bug where downloading models from Hugging Face could freeze
- Python package upgrades
  - huggingface-hub 0.19.4 -> 0.20.2
  - bigframes 0.17.0 -> 0.18.0

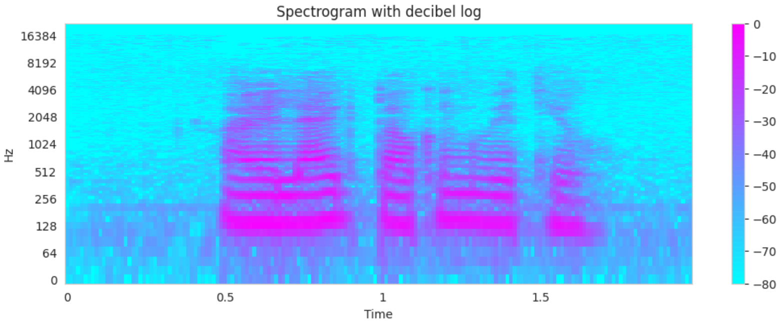
2023-12-18

- Expanded access to AI coding has arrived in Colab across 175 locales for al tiers of Colab users
- Improvements to display of ML-based inline completions (for eligible Pro/Pro+ users)
- Started a series of notebooks highlighting Gemini API capabilities
- Enable ⌘/Ctrl+L to select the full line in an editor
- Fixed bug where we weren't correctly formatting output from multiple execution results
- Python package upgrades
  - CUDA 11.8 to CUDA 12.2
  - tensorflow 2.14.0 -> 2.15.0
  - tensorboard 2.14.0 -> 2.15.0
  - keras 2.14.0 -> 2.15.0
  - Nvidia drivers 525.105.17 -> 535.104.05
  - tensorflow-gcs-config 2.14.0 -> 2.15.0
  - bigframes 0.13.0 -> 0.17.0
  - geemap 0.28.2 -> 0.29.6
  - pyarrow 9.0.0 -> 10.0.1
  - google-generativeai 0.2.2 -> 0.3.1
  - jax 0.4.20 —> 0.4.23
  - jaxlib 0.4.20 —> 0.4.23
- Python package inclusions
  - kagglehub 0.1.4
  - google-cloud-aiplatform 1.38.1

2023-11-27

- Removed warning when calling `await` to make it render as code
- Added "Run selection" to the cell context menu
- Added highlighting for the `%%python` cell magic
- Launched AI coding features for Pro/Pro+ users in more locales
- Python package upgrades
  - bigframes 0.12.0 -> 0.13.0
- Python package inclusions





```
<Figure size 640x480 with 0 Axes>
```

```
melspec = librosa.feature.melspectrogram(y=audio_data, sr=sample_r
melspec_db = librosa.amplitude_to_db(melspec, ref=np.max)

plt.figure(figsize=(12,4))
lplt.specshow(melspec, sr=sr, x_axis='time', y_axis='hz')
plt.colorbar()
plt.title("Spectrogram with amplitude")
plt.show()

plt.figure(figsize=(12,4))
lplt.specshow(melspec_db, sr=sr, x_axis='time', y_axis='log', cmap
plt.colorbar()
plt.title("Spectrogram with decibel log")
plt.show()
```

- transformers 4.35.2
- google-generativeai 0.2.2

2023-11-08

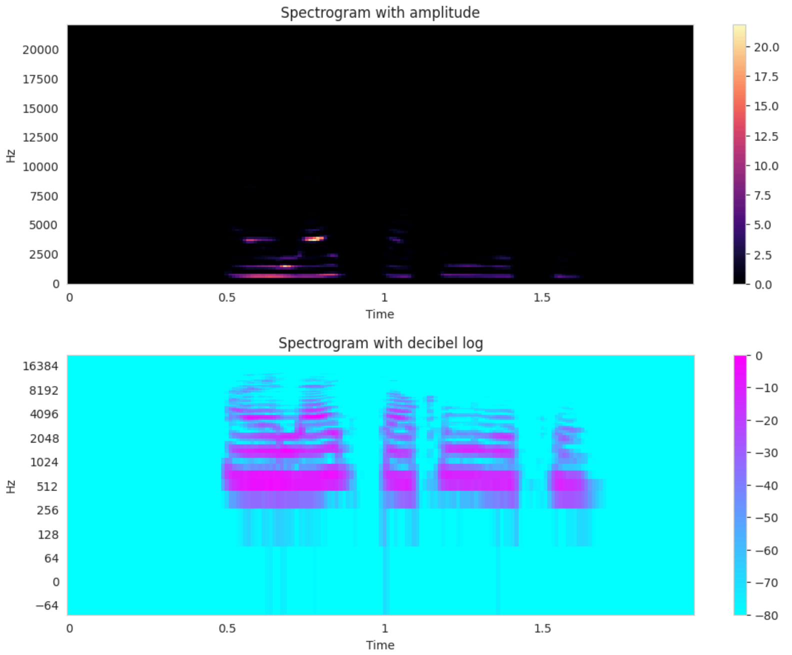
- Launched Secrets, for safe storage of private keys on Colab (tweet)
- Fixed issue where TensorBoard would not load (#3990)
- Python package upgrades
  - lightgbm 4.0.0 -> 4.1.0
  - bigframes 0.10.0 -> 0.12.0
  - bokeh 3.2.2 -> 3.3.0
  - duckdb 0.8.1 -> 0.9.1
  - numba 0.56.4 -> 0.58.1
  - tweepy 4.13.0 -> 4.14.0
  - jax 0.4.16 -> 0.4.20
  - jaxlib 0.4.16 -> 0.4.20

2023-10-23

- Updated the **Open notebook** dialog for better usability and support for smaller screen sizes
- Added smart paste support for data from Google Sheets for R notebooks
- Enabled showing release notes in a tab
- Launched AI coding features for Pro/Pro+ users in Australia AU Canada CA India IN and Japan JP (tweet)
- Python package upgrades
  - earthengine-api 0.1.357 -> 0.1.375
  - flax 0.7.2 -> 0.7.4
  - geemap 0.27.4 -> 0.28.2
  - jax 0.4.14 -> 0.4.16
  - jaxlib 0.4.14 -> 0.4.16
  - keras 2.13.1 -> 2.14.0
  - tensorboard 2.13.0 -> 2.14.1
  - tensorflow 2.13.0 -> 2.14.0
  - tensorflow-gcs-config 2.13.0 -> 2.14.0
  - tensorflow-hub 0.14.0 -> 0.15.0
  - tensorflow-probability 0.20.1 -> 0.22.0
  - torch 2.0.1 -> 2.1.0
  - torchaudio 2.0.2 -> 2.1.0
  - torchtext 0.15.2 -> 0.16.0
  - torchvision 0.15.2 -> 0.16.0
  - xgboost 1.7.6 -> 2.0.0
- Python package inclusions
  - bigframes 0.10.0
  - malloy 2023.1056

2023-09-22

- Added the ability to scope an AI generated suggestion to a specific Pandas dataframe (tweet)
- Added Colab link previews to Docs (tweet)
- Added smart paste support for data from Google Sheets
- Increased font size of dropdowns in interactive forms
- Improved rendering of the notebook when printing
- Python package upgrades
  - tensorflow 2.12.0 -> 2.13.0
  - tensorboard 2.12.3 -> 2.13.0
  - keras 2.12.0 -> 2.13.1



```
<Figure size 640x480 with 0 Axes>
```

Data Preparation

```
<Figure size 640x480 with 0 Axes>
```

Encode Genre Label

```
<Figure size 640x480 with 0 Axes>
```

```
# map labels to index
label_index = dict()
index_label = dict()
for i, x in enumerate(df_base['Class'].unique()):
    label_index[x] = i
    index_label[i] = x
print(label_index)
print(index_label)
```

```
{'Negative': 0, 'Neutral': 1, 'Positive': 2}
{0: 'Negative', 1: 'Neutral', 2: 'Positive'}
<Figure size 640x480 with 0 Axes>
```

```
# update labels in df to index
train_labels_idx = [label_index[l] for l in train_labels]
train_labels_idx[::10]
```

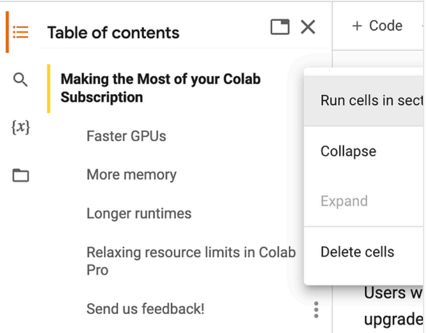
  - tensorflow-gcs-config 2.12.0 -> 2.13.
  - scipy 1.10.1-> 1.11.2
  - cython 0.29.6 -> 3.0.2
- Python package inclusions
  - geemap 0.26.0

2023-08-18

- Added "Change runtime type" to the menu in the connection button
- Improved auto-reconnection to an already running notebook (#3764)
- Increased the specs of our highmem machines for Pro users
- Fixed `add-apt-repository` command on Ubuntu 22.04 runtime (#3867)
- Python package upgrades
  - bokeh 2.4.3 -> 3.2.2
  - cmake 3.25.2 -> 3.27.2
  - cryptography 3.4.8 -> 41.0.3
  - dask 2022.12.1 -> 2023.8.0
  - distributed 2022.12.1 -> 2023.8.0
  - earthengine-api 0.1.358 -> 0.1.364
  - flax 0.7.0 -> 0.7.2
  - ipython-sql 0.4.0 -> 0.5.0
  - jax 0.4.13 -> 0.4.14
  - jaxlib 0.4.13 -> 0.4.14
  - lightgbm 3.3.5 -> 4.0.0
  - mkl 2019.0 -> 2023.2.0
  - notebook 6.4.8 -> 6.5.5
  - numpy 1.22.4 -> 1.23.5
  - opencv-python 4.7.0.72 -> 4.8.0.76
  - pillow 8.4.0 -> 9.4.0
  - plotly 5.13.1 -> 5.15.0
  - prettytable 0.7.2 -> 3.8.0
  - pytensor 2.10.1 -> 2.14.2
  - spacy 3.5.4 -> 3.6.1
  - statsmodels 0.13.5 -> 0.14.0
  - xarray 2022.12.0 -> 2023.7.0
- Python package inclusions
  - PyDrive2 1.6.3

2023-07-21

- Launched auto-plotting for dataframes, available using the chart button that shows up alongside datatables (post)
- Added a menu to the table of contents to support running a section or collapsing/expanding sections (post)



- Added an option to automatically run the first cell or section, available under Edit -> Notebook settings (post)

```
[2, 2, 2, 1, 1, 1, 2, 1, 1, 1, 1, 1, 0, 0, 0, 0, 2, 0, 0, 0,
0, 2, 2, 2, 2]
```

Split Train & Test Sets

```
# shuffle samples
df_shuffle = df_base.sample(frac=1, random_state=seed).reset_index


# remove irrelevant columns
df_shuffle.drop(['Filename'], axis=1, inplace=True)
df_y = df_shuffle.pop('Class')

# split into train dev and test
y_train, y_test = skms.train_test_split(df_y, train_size=0.8, ranc


print(f"Train set has {y_train.shape[0]} records out of {len(df_sh
print(f"Test set has {y_test.shape[0]} records out of {len(df_shuf
```

```
Train set has 200 records out of 250 which is 80%
Test set has 50 records out of 250 which is 20%
```

```
# stratified split check
print(y_train.value_counts())
print(y_test.value_counts())
```

```
Class
Negative    69
Positive    66
Neutral     65
Name: count, dtype: int64
Class
Negative    18
Neutral     16
Positive    16
Name: count, dtype: int64
```

```
# divide train_data into X_train and X_test
X_train = train_data[y_train.index.tolist(), :, :]
X_test = train_data[y_test.index.tolist(), :, :]
X_test.shape
```

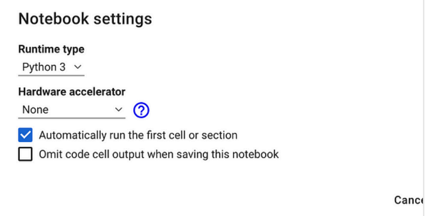
```
(50, 128, 128)
```

```
# divide train_data_img into X_train_img and X_test_img
X_train_img = train_data_img[y_train.index.tolist(), :, :]
X_test_img = train_data_img[y_test.index.tolist(), :, :]
X_test_img.shape
```

```
(50, 128, 128, 3)
```

```
y_train = np.array([train_labels_idx[x] for x in y_train.index.tol
y_test = np.array([train_labels_idx[x] for x in y_test.index.tolis
y_train[::10]
```

```
array([1, 2, 2, 2, 2, 0, 2, 0, 1, 1, 1, 1, 2, 0, 0, 2, 1, 0,
0, 2])
```



Notebook settings

Runtime type

Python 3

Hardware accelerator

None

☑ Automatically run the first cell or section

☐ Omit code cell output when saving this notebook

- Launched Pro/Pro+ to Algeria, Argentina, Chile, Ecuador, Egypt, Ghana, Kenya, Malaysia, Nepal, Nigeria, Peru, Rwanda, Saudi Arabia, South Africa, Sri Lanka, Tunisia, and Ukraine (tweet)
- Added a command, "Toggle tab moves focus" for toggling tab trapping in the editor (Tools -> Command palette, "Toggle tab moves focus")
- Fixed issue where `files.upload()` was sometimes returning an incorrect filename (#1550)
- Fixed f-string syntax highlighting bug (#3802)
- Disabled ambiguous characters highlighting for commonly used LaTeX characters (#3648)
- Upgraded Ubuntu from 20.04 LTS to 22.04 LTS
- Updated the Colab Marketplace VM image
- Python package upgrades:
  - autograd 1.6.1 -> 1.6.2
  - drivefs 76.0 -> 77.0
  - flax 0.6.11 -> 0.7.0
  - earthengine-api 0.1.357 -> 0.1.358
  - GDAL 3.3.2->3.4.3
  - google-cloud-bigquery-storage 2.20.0 -> 2.22.2
  - gspread-dataframe 3.0.8 -> 3.3.1
  - holidays 0.27.1 -> 0.29
  - jax 0.4.10 -> jax 0.4.13
  - jaxlib 0.4.10 -> jax 0.4.13
  - jupyterlab-widgets 3.0.7 -> 3.0.8
  - nbformat 5.9.0 -> 5.9.1
  - opencv-python-headless 4.7.0.72 -> 4.8.0.74
  - pygame 2.4.0 -> 2.5.0
  - spacy 3.5.3 -> 3.5.4
  - SQLAlchemy 2.0.16 -> 2.0.19
  - tabulate 0.8.10 -> 0.9.0
  - tensorflow-hub 0.13.0 -> 0.14.0

2023-06-23

- Launched AI coding features to subscribed users starting with Pro+ users in the US (tweet, post)
- Added the Kernel Selector in the Notebook Settings (tweet)
- Fixed double space trimming issue in markdown #3766
- Fixed run button indicator not always centered #3609
- Fixed inconsistencies for automatic indentation on multi-line #3697
- Upgraded Python from 3.10.11 to 3.10.12
- Python package updates:
  - duckdb 0.7.1 -> 0.8.1
  - earthengine-api 0.1.350 -> 0.1.357
  - flax 0.6.9 -> 0.6.11

Scale the Features

```
# scale features
scaler = skp.MinMaxScaler()
X_train = scaler.fit_transform(X_train.reshape(-1, X_train.shape[-
X_test = scaler.transform(X_test.reshape(-1, X_test.shape[-1])).re
test_data = scaler.transform(test_data.reshape(-1, test_data.shape
print(X_train.shape)
```

```
(200, 128, 128)
```

Model Building

```
import tensorflow as tf
print("TF version:-", tf.__version__)
import keras as k
from keras import backend as K
tf.random.set_seed(seed)
```

```
TF version:- 2.15.0
```

```
bestModelPath = './best_model.hdf5'
ACCURACY_THRESHOLD = 0.98

class myCallback(k.callbacks.Callback):
    def on_epoch_end(self, epoch, logs={}):
        if(logs.get('val_accuracy') > ACCURACY_THRESHOLD):
            print("\n\nStopping training as we have reached %2.2f%
            self.model.stop_training = True

acc_callback = myCallback()


def trainModel(model, epochs, optimizer, vb=1):
    cbs = [#k.callbacks.ReduceLROnPlateau(patience=5, verbose=1),
           k.callbacks.ModelCheckpoint(filepath=bestModelPath, mor
    batch_size = 64
    callback = myCallback()
    model.compile(optimizer=optimizer,
                  loss='sparse_categorical_crossentropy',
                  metrics='accuracy'
    )
    return model.fit(X_train, y_train,
#                      validation_data=(X_test, y_test),
                     epochs=epochs, verbose=vb,
                     validation_split=0.2,
                     batch_size=batch_size, callbacks=cbs)

def plotHistory(history):
    print("Max. Validation Accuracy",max(history.history["val_accu
    pd.DataFrame(history.history).plot(figsize=(12,6))
    plt.show()


model_1 = k.models.Sequential([
    k.layers.Conv1D(256, 8, padding='same', activation='relu', inp
#     k.layers.Conv1D(256, 8, padding='same', activation='relu'),
    k.layers.BatchNormalization(),
    k.layers.Dropout(0.2),
    k.layers.MaxPooling1D(pool_size=(8)),
```

  - google-cloud-bigquery 3.9.0 -> 3.10.0
  - google-cloud-bigquery-storage 2.19.1 -> 2.20.0
  - grpcio 1.54.0 -> 1.56.0
  - holidays 0.25 -> 0.27.1
  - nbformat 5.8.0 -> 5.9.0
  - prophet 1.1.3 -> 1.1.4
  - pydata-google-auth 1.7.0 -> 1.8.0
  - spacy 3.5.2 -> 3.5.3
  - tensorboard 2.12.2 -> 2.12.3
  - xgboost 1.7.5 -> 1.7.6
- Python package inclusions:
  - gcsfs 2023.6.0
  - geopandas 0.13.2
  - google-cloud-bigquery-connection 1.12.0
  - google-cloud-functions 1.13.0
  - grpc-google-iam-v1 0.12.6
  - multidict 6.0.4
  - tensorboard-data-server 0.7.1

2023-06-02

- Released the new site colab.google
- Published Colab's Docker runtime image to us-docker.pkg.dev/colab-images/public/runtime (tweet, instructions)
- Launched support for Google children accounts (tweet)
- Launched DagsHub integration (tweet, post)
- Upgraded to Monaco Editor Version 0.37.1
- Fixed various Vim keybinding bugs
- Fixed issue where the N and P letters sometimes couldn't be typed (#3664)
- Fixed rendering support for compositional inputs (#3660, #3679)
- Fixed lag in notebooks with lots of cells (#3676)
- Improved support for R by adding a Runtime type notebook setting (Edit -> Notebook settings)
- Improved documentation for connecting to a local runtime (Connect -> Connect to a local runtime)
- Python package updates:
  - holidays 0.23 -> 0.25
  - jax 0.4.8 -> 0.4.10
  - jaxlib 0.4.8 -> 0.4.10
  - pip 23.0.1 -> 23.1.2
  - tensorflow-probability 0.19.0 -> 0.20.1
  - torch 2.0.0 -> 2.0.1
  - torchaudio 2.0.1 -> 2.0.2
  - torchdata 0.6.0 -> 0.6.1
  - torchtext 0.15.1 -> 0.15.2
  - torchvision 0.15.1 -> 0.15.2
  - tornado 6.2 -> 6.3.1

2023-05-05

- Released GPU type selection for paid users, allowing them to choose a preferred NVidia GPU
- Upgraded R from 4.2.3 to 4.3.0
- Upgraded Python from 3.9.16 to 3.10.11
- Python package updates:
  - attrs 22.2.0 -> attrs 23.1.0

```
    k.layers.Conv1D(128, 8, padding='same', activation='relu'),
#     k.layers.Conv1D(128, 8, padding='same', activation='relu'),
#     k.layers.Conv1D(128, 8, padding='same', activation='relu'),
    k.layers.BatchNormalization(),
    k.layers.Dropout(0.2),
    k.layers.MaxPooling1D(pool_size=(5)),
#     k.layers.Conv1D(64, 8, padding='same', activation='relu'),
    k.layers.Conv1D(64, 8, padding='same', activation='relu'),
    k.layers.BatchNormalization(),
    k.layers.Flatten(),
#     k.layers.Dense(64, activation='relu'),
    k.layers.Dense(len(index_label), activation='softmax'),
])
print(model_1.summary())
model_1_history = trainModel(model=model_1, epochs=50, optimizer='
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Para
=================================================================
 conv1d (Conv1D)             (None, 128, 256)          2624

 batch_normalization (Batch  (None, 128, 256)          1024
 Normalization)

 dropout (Dropout)           (None, 128, 256)          0

 max_pooling1d (MaxPooling1  (None, 16, 256)           0
 D)

 conv1d_1 (Conv1D)           (None, 16, 128)           2622

 batch_normalization_1 (Bat  (None, 16, 128)           512
 chNormalization)

 dropout_1 (Dropout)         (None, 16, 128)           0

 max_pooling1d_1 (MaxPoolin  (None, 3, 128)            0
 g1D)

 conv1d_2 (Conv1D)           (None, 3, 64)             6560

 batch_normalization_2 (Bat  (None, 3, 64)             256
 chNormalization)

 flatten (Flatten)           (None, 192)               0

 dense (Dense)               (None, 3)                 579

=================================================================
Total params: 592643 (2.26 MB)
Trainable params: 591747 (2.26 MB)
Non-trainable params: 896 (3.50 KB)
_________________________________________________________________
None

Epoch 1: val_loss improved from inf to 1.03816, saving mode

Epoch 2: val_loss improved from 1.03816 to 1.01395, saving

Epoch 3: val_loss improved from 1.01395 to 0.96782, saving

Epoch 4: val_loss improved from 0.96782 to 0.93066, saving

Epoch 5: val_loss improved from 0.93066 to 0.89464, saving

Epoch 6: val_loss improved from 0.89464 to 0.86782, saving

Epoch 7: val_loss improved from 0.86782 to 0.84906, saving
```

- earthengine-api 0.1.349 -> earthengine-api 0.1.350
- flax 0.6.8 -> 0.6.9
- grpcio 1.53.0 -> 1.54.0
- nbclient 0.7.3 -> 0.7.4
- tensorflow-datasets 4.8.3 -> 4.9.2
- termcolor 2.2.0 -> 2.3.0
- zict 2.2.0 -> 3.0.0

2023-04-14

- Python package updates:
  - google-api-python-client 2.70.0 -> 2.84.0
  - google-auth-oauthlib 0.4.6 -> 1.0.0
  - google-cloud-bigquery 3.4.2 -> 3.9.0
  - google-cloud-datastore 2.11.1 -> 2.15.1
  - google-cloud-firestore 2.7.3 -> 2.11.0
  - google-cloud-language 2.6.1 -> 2.9.1
  - google-cloud-storage 2.7.0 -> 2.8.0
  - google-cloud-translate 3.8.4 -> 3.11.1
  - networkx 3.0 -> 3.1
  - notebook 6.3.0 -> 6.4.8
  - jax 0.4.7 -> 0.4.8
  - pandas 1.4.4 -> 1.5.3
  - spacy 3.5.1 -> 3.5.2
  - SQLAlchemy 1.4.47 -> 2.0.9
  - xgboost 1.7.4 -> 1.7.5

2023-03-31

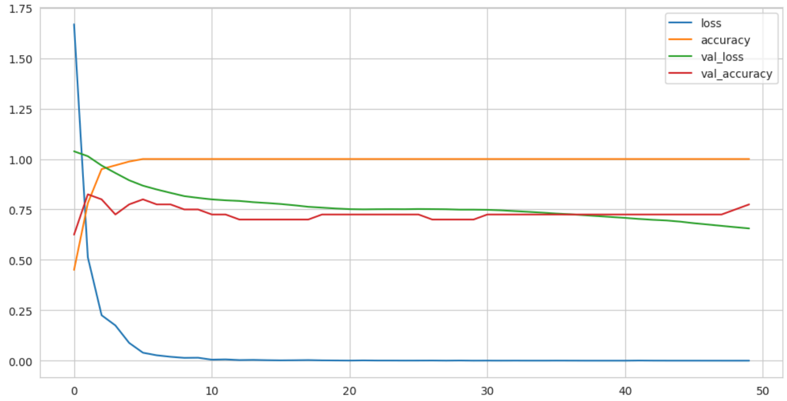
- Improve bash ! syntax highlighting (GitHub issue)
- Fix bug where VIM keybindings weren't working in the file editor
- Upgraded R from 4.2.2 to 4.2.3
- Python package updates:
  - arviz 0.12.1 --> 0.15.1
  - astropy 4.3.1 --> 5.2.2
  - dopamine-rl 1.0.5 --> 4.0.6
  - gensim 3.6.0 --> 4.3.1
  - ipykernel 5.3.4 -> 5.5.6
  - ipython 7.9.0 -> 7.34.0
  - jax 0.4.4 -> 0.4.7
  - jaxlib 0.4.4 -> 0.4.7
  - jupyter_core 5.2.0 -> 5.3.0
  - keras 2.11.0 -> 2.12.0
  - lightgbm 2.2.3 -> 3.3.5
  - matplotlib 3.5.3 -> 3.7.1
  - nltk 3.7 -> 3.8.1
  - opencv-python 4.6.0.66 -> 4.7.0.72
  - plotly 5.5.0 -> 5.13.1
  - pymc 4.1.4 -> 5.1.2
  - seaborn 0.11.2 -> 0.12.2
  - spacy 3.4.4 -> 3.5.1
  - sympy 1.7.1 -> 1.11.1
  - tensorboard 2.11.2 -> 2.12.0
  - tensorflow 2.11.0 -> 2.12.0
  - tensorflow-estimator 2.11.0 -> 2.12.0
  - tensorflow-hub 0.12.0 -> 0.13.0
  - torch 1.13.1 -> 2.0.0
  - torchaudio 0.13.1 -> 2.0.1
  - torchtext 0.14.1 -> 0.15.1
  - torchvision 0.14.1 -> 0.15.1

2023-03-10

```
Epoch 8: val_loss improved from 0.84906 to 0.83227, saving
```

```
plotHistory(model_1_history)
```

```
Max. Validation Accuracy 0.824999988079071
```



```
# evaluation
test_loss, test_acc  = k.models.load_model(bestModelPath).evaluate
print("The test Loss is :",test_loss)
print("The test Accuracy is :",test_acc*100)
```

```
1/1 [==============================] - 1s 926ms/step - loss: 0
The test Loss is : 0.7504545450210571
The test Accuracy is : 62.00000047683716
```

```
model_2 = k.models.Sequential([
    k.layers.Conv1D(256, 5, activation='relu', input_shape=(X_trai
    k.layers.BatchNormalization(),
    k.layers.Dropout(0.3),
    k.layers.MaxPooling1D(pool_size=(2)),
    k.layers.Conv1D(128, 3, activation='relu'),
    k.layers.BatchNormalization(),
    k.layers.Dropout(0.3),
    k.layers.MaxPooling1D(pool_size=(3)),
    k.layers.Conv1D(64, 3, activation='relu'),
    k.layers.BatchNormalization(),
    k.layers.Flatten(),
    k.layers.Dense(32, activation='relu'),
    k.layers.Dense(len(index_label), activation='softmax'),
])
print(model_2.summary())
model_2_history = trainModel(model=model_2, epochs=100, optimizer=
```

```
Model: "sequential_1"
_____________________________________________________________
 Layer (type)                Output Shape              Para
=============================================================
 conv1d_3 (Conv1D)           (None, 124, 256)          1640
```

- Added the Colab editor shortcuts example notebook
- Fixed triggering of @-mention and email autocomplete for large comments (GitHub issue)
- Added View Resources to the Runtime menu
- Made file viewer images fit the view by default, resizing to original size on click
- When in VIM mode, enable copy as well as allowing propagation to monaco-vim to escape visual mode (GitHub issue)
- Upgraded CUDA 11.6.2 -> 11.8.0 and cuDNN 8.4.0.27 -> 8.7.0.84
- Upgraded Nvidia drivers 525.78.01 -> 530.30.02
- Upgraded Python 3.8.10 -> 3.9.16
- Python package updates:
    - beautifulsoup4 4.6.3 -> 4.9.3
    - bokeh 2.3.3 -> 2.4.3
    - debugpy 1.0.0 -> 1.6.6
    - Flask 1.1.4 -> 2.2.3
    - jax 0.3.25 -> 0.4.4
    - jaxlib 0.3.25 -> 0.4.4
    - Jinja2 2.11.3 -> 3.1.2
    - matplotlib 3.2.2 -> 3.5.3
    - nbconvert 5.6.1 -> 6.5.4
    - pandas 1.3.5 -> 1.4.4
    - pandas-datareader 0.9.0 -> 0.10.0
    - pandas-profiling 1.4.1 -> 3.2.0
    - Pillow 7.1.2 -> 8.4.0
    - plotnine 0.8.0 -> 0.10.1
    - scikit-image 0.18.3 -> 0.19.3
    - scikit-learn 1.0.2 -> 1.2.2
    - scipy 1.7.3 -> 1.10.1
    - setuptools 57.4.0 -> 63.4.3
    - sklearn-pandas 1.8.0 -> 2.2.0
    - statsmodels 0.12.2 -> 0.13.5
    - urllib3 1.24.3 -> 1.26.14
    - Werkzeug 1.0.1 -> 2.2.3
    - wrapt 1.14.1 -> 1.15.0
    - xgboost 0.90 -> 1.7.4
    - xlrd 1.2.0 -> 2.0.1

2023-02-17

- Show graphs of RAM and disk usage in notebook toolbar
- Copy cell links directly to the clipboard instead of showing a dialog when clicking on the link icon in the cell toolbar
- Updated the Colab Marketplace VM image
- Upgraded CUDA to 11.6.2 and cuDNN to 8.4.0.27
- Python package updates:
    - tensorflow 2.9.2 -> 2.11.0
    - tensorboard 2.9.1 -> 2.11.2
    - keras 2.9.0 -> 2.11.0
    - tensorflow-estimator 2.9.0 -> 2.11.0
    - tensorflow-probability 0.17.0 -> 0.19.0
    - tensorflow-gcs-config 2.9.0 -> 2.11.0
    - earthengine-api 0.1.339 -> 0.1.341
    - flatbuffers 1.12 -> 23.1.21
    - platformdirs 2.6.2 -> 3.0.0
    - pydata-google-auth 1.6.0 -> 1.7.0
    - python-utils 3.4.5 -> 3.5.2
    - tenacity 8.1.0 -> 8.2.1
    - tifffile 2023.1.23.1 -> 2023.2.3

```
 batch_normalization_3 (Bat  (None, 124, 256)         1024
 chNormalization)

 dropout_2 (Dropout)         (None, 124, 256)         0

 max_pooling1d_2 (MaxPoolin  (None, 62, 256)          0
 g1D)

 conv1d_4 (Conv1D)           (None, 60, 128)          9843

 batch_normalization_4 (Bat  (None, 60, 128)          512
 chNormalization)

 dropout_3 (Dropout)         (None, 60, 128)          0

 max_pooling1d_3 (MaxPoolin  (None, 20, 128)          0
 g1D)

 conv1d_5 (Conv1D)           (None, 18, 64)           2464

 batch_normalization_5 (Bat  (None, 18, 64)           256
 chNormalization)

 flatten_1 (Flatten)         (None, 1152)             0

 dense_1 (Dense)             (None, 32)               3689

 dense_2 (Dense)             (None, 3)                99

============================================================
Total params: 325955 (1.24 MB)
Trainable params: 325059 (1.24 MB)
Non-trainable params: 896 (3.50 KB)
____________________________________________________________
None

Epoch 1: val_loss improved from inf to 1.08570, saving mode

Epoch 2: val_loss improved from 1.08570 to 1.07150, saving

Epoch 3: val_loss improved from 1.07150 to 1.03410, saving

Epoch 4: val_loss improved from 1.03410 to 0.98863, saving

Epoch 5: val_loss improved from 0.98863 to 0.94864, saving

Epoch 6: val_loss improved from 0.94864 to 0.92635, saving

Epoch 7: val_loss improved from 0.92635 to 0.90491, saving

Epoch 8: val_loss improved from 0.90491 to 0.87945, saving
```

```
plotHistory(model_2_history)
```

- notebook 5.7.16 -> 6.3.0
- tornado 6.0.4 -> 6.2
- aiohttp 3.8.3 -> 3.8.4
- charset-normalizer 2.1.1 -> 3.0.1
- fastai 2.7.0 -> 2.7.1
- soundfile 0.11.0 -> 0.12.1
- typing-extensions 4.4.0 -> 4.5.0
- widgetsnbextension 3.6.1 -> 3.6.2
- pydantic 1.10.4 -> 1.10.5
- zipp 3.12.0 -> 3.13.0
- numpy 1.21.6 -> 1.22.4
- drivefs 66.0 -> 69.0
- gdal 3.0.4 -> 3.3.2 GitHub issue

- Added libudunits2-dev for smoother R package installs GitHub issue

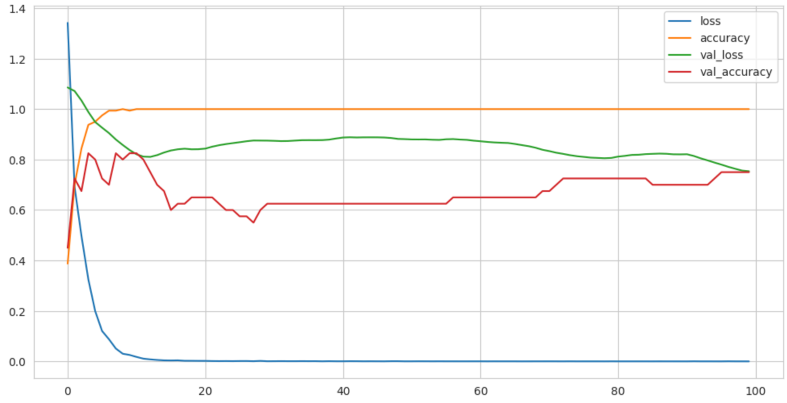
2023-02-03

- Improved tooltips for pandas series to show common statistics about the series object
- Made the forms dropdown behave like an autocomplete box when it allows input
- Updated the nvidia driver from 460.32.03 to 510.47.03
- Python package updates:
  - absl-py 1.3.0 -> 1.4.0
  - bleach 5.0.1 -> 6.0.0
  - cachetools 5.2.1 -> 5.3.0
  - cmdstanpy 1.0.8 -> 1.1.0
  - dnspython 2.2.1 -> 2.3.0
  - fsspec 2022.11.0 -> 2023.1.0
  - google-cloud-bigquery-storage 2.17.0 -> 2.18.1
  - holidays 0.18 -> 0.19
  - jupyter-core 5.1.3 -> 5.2.0
  - packaging 21.3 -> 23.0
  - prometheus-client 0.15.0 -> 0.16.0
  - pyct 0.4.8 -> 0.5.0
  - pydata-google-auth 1.5.0 -> 1.6.0
  - python-slugify 7.0.0 -> 8.0.0
  - sqlalchemy 1.4.46 -> 2.0.0
  - tensorflow-io-gcs-filesystem 0.29.0 -> 0.30.0
  - tifffile 2022.10.10 -> 2023.1.23.1
  - zipp 3.11.0 -> 3.12.0
  - Pinned sqlalchemy to version 1.4.46

2023-01-12

- Added support for @-mention and email autocomplete in comments
- Improved errors when GitHub notebooks can't be loaded
- Increased color contrast for colors used for syntax highlighting in the code editor
- Added terminal access for custom GCE VM runtimes
- Upgraded Ubuntu from 18.04 LTS to 20.04 LTS (GitHub issue)
- Python package updates:
  - GDAL 2.2.2 -> 2.2.3.
  - NumPy from 1.21.5 to 1.21.6.
  - attrs 22.1.0 -> 22.2.0
  - chardet 3.0.4 -> 4.0.0
  - cloudpickle 1.6.0 -> 2.2.0
  - filelock 3.8.2 -> 3.9.0
  - google-api-core 2.8.2 -> 2.11.0
  - google-api-python-client 1.12.11 -> 2.70.0
  - google-auth-httplib2 0.0.3 -> 0.1.0

```
Max. Validation Accuracy 0.824999988079071
```



```
# evaluation
test_loss, test_acc  = k.models.load_model(bestModelPath).evaluate
print("The test Loss is :",test_loss)
print("The test Accuracy is :",test_acc*100)
```

```
1/1 [==============================] - 0s 316ms/step - loss: 0
The test Loss is : 0.7409237623214722
The test Accuracy is : 66.00000262260437
```

```
model_3 = k.models.Sequential([
    k.layers.Bidirectional(k.layers.LSTM(256, return_sequences=Tru

    k.layers.Bidirectional(k.layers.LSTM(128, return_sequences=Fal

    k.layers.Dense(64, activation='relu'),
    k.layers.Dropout(0.2),
    k.layers.Dense(64, activation='relu'),
    k.layers.Dropout(0.2),
    k.layers.Dense(32, activation='relu'),
    k.layers.Dense(len(index_label), activation='softmax'),
])
print(model_3.summary())
model_3_history = trainModel(model=model_3, epochs=100, optimizer=
```

```
Model: "sequential_2"
_________________________________________________________________
 Layer (type)                Output Shape              Para
=================================================================
 bidirectional (Bidirection  (None, 128, 512)          7884
 al)

 bidirectional_1 (Bidirecti  (None, 256)               6563
 onal)

 dense_3 (Dense)             (None, 64)                1644

 dropout_4 (Dropout)         (None, 64)                0

 dense_4 (Dense)             (None, 64)                4160
```

- google-cloud-bigquery 3.3.5 -> 3.4.1
- google-cloud-datastore 2.9.0 -> 2.11.0
- google-cloud-firestore 2.7.2 -> 2.7.3
- google-cloud-storage 2.5.0 -> 2.7.0
- holidays 0.17.2 -> holidays 0.18
- importlib-metadata 5.2.0 -> 6.0.0
- networkx 2.8.8 -> 3.0
- opencv-python-headless 4.6.0.66 -> 4.7.0.68
- pip 21.1.3 -> 22.04
- pip-tools 6.2.0 -> 6.6.2
- prettytable 3.5.0 -> 3.6.0
- requests 2.23.0 -> 2.25.1
- termcolor 2.1.1 -> 2.2.0
- torch 1.13.0 -> 1.13.1
- torchaudio 0.13.0 -> 0.13.1
- torchtext 0.14.0-> 0.14.1
- torchvision 0.14.0 -> 0.14.1

2022-12-06

- Made fallback runtime version available until mid-December (GitHub issue)
- Upgraded to Python 3.8 (GitHub issue)
- Python package updates:
    - jax from 0.3.23 to 0.3.25, jaxlib from 0.3.22 to 0.3.25
    - pyarrow from 6.0.1 to 9.0.0
    - torch from 1.12.1 to 1.13.0
    - torchaudio from 0.12.1 to 0.13.0
    - torchvision from 0.13.1 to 0.14.0
    - torchtext from 0.13.1 to 0.14.0
    - xlrd from 1.1.0 to 1.2.0
    - DriveFS from 62.0.1 to 66.0.3
- Made styling of markdown tables in outputs match markdown tables in text cells
- Improved formatting for empty interactive table rows
- Fixed syntax highlighting for variables with names that contain Python keywords (GitHub issue)

2022-11-11

- Added more dark editor themes for Monaco (when in dark mode, "Editor colorization" appears as an option in the Editor tab of the Tools → Settings dialog)
- Fixed bug where collapsed forms were deleted on mobile GitHub issue
- Python package updates:
    - rpy2 from 3.4.0 to 3.5.5 (GitHub issue)
    - notebook from 5.5.0 to 5.7.16
    - tornado from 5.1.1 to 6.0.4
    - tensorflow_probability from 0.16.0 to 0.17.0
    - pandas-gbq from 0.13.3 to 0.17.9
    - protobuf from 3.17.3 to 3.19.6
    - google-api-core[grpc] from 1.31.5 to 2.8.2
    - google-cloud-bigquery from 1.21.0 to 3.3.5
    - google-cloud-core from 1.0.1 to 2.3.2
    - google-cloud-datastore from 1.8.0 to 2.9.0

```
 dropout_5 (Dropout)         (None, 64)               0

 dense_5 (Dense)             (None, 32)               2080

 dense_6 (Dense)             (None, 3)                99

=========================================================
Total params: 1467651 (5.60 MB)
Trainable params: 1467651 (5.60 MB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________
None

Epoch 1: val_loss improved from inf to 1.09762, saving mode

Epoch 2: val_loss improved from 1.09762 to 1.05798, saving

Epoch 3: val_loss did not improve from 1.05798

Epoch 4: val_loss improved from 1.05798 to 1.02973, saving

Epoch 5: val_loss did not improve from 1.02973

Epoch 6: val_loss improved from 1.02973 to 1.00481, saving

Epoch 7: val_loss did not improve from 1.00481

Epoch 8: val_loss improved from 1.00481 to 0.96885, saving

Epoch 9: val_loss improved from 0.96885 to 0.87561, saving

Epoch 10: val_loss did not improve from 0.87561

Epoch 11: val_loss did not improve from 0.87561

Epoch 12: val_loss improved from 0.87561 to 0.78955, saving

Epoch 13: val_loss did not improve from 0.78955

Epoch 14: val_loss did not improve from 0.78955
```

```
plotHistory(model_3_history)
```

```
# evaluation
test_loss, test_acc  = k.models.load_model(bestModelPath).evaluate
print("The test Loss is :",test_loss)
print("The test Accuracy is :",test_acc*100)
```

```
1/1 [==============================] - 3s 3s/step - loss: 0.66
The test Loss is : 0.6610075235366821
The test Accuracy is : 83.99999737739563
```

```
# make features 3D with last dim as 1 for 1DConv
X_train = np.expand_dims(X_train, axis=3)
X_test = np.expand_dims(X_test, axis=3)
X_train.shape
```

```
(200, 128, 128, 1)
```

```
model_4 = k.models.Sequential([
    k.layers.Conv2D(256, (5,5), activation='relu', input_shape=(X_
    k.layers.BatchNormalization(),
    k.layers.MaxPooling2D(pool_size=(2)),
    k.layers.Dropout(0.3),
```

  - google-cloud-firestore from 1.7.0 to 2.7.2
  - google-cloud-language from 1.2.0 to 2.6.1
  - google-cloud-storage from 1.18.0 to 2.5.0
  - google-cloud-translate from 1.5.0 to 3.8.4

2022-10-21

- Launched a single-click way to get from BigQuery to Colab to further explore query results (announcement)
- Launched Pro, Pro+, and Pay As You Go to 19 additional countries: Austria, Belgium, Bulgaria, Croatia, Cyprus, Czechia, Denmark, Estonia, Finland, Greece, Hungary, Latvia, Lithuania, Norway, Portugal, Romania, Slovakia, Slovenia, and Sweden (tweet)
- Updated jax from 0.3.17 to 0.3.23, jaxlib from 0.3.15 to 0.3.22, TensorFlow from 2.8.2 to 2.9.2, CUDA from 11.1 to 11.2, and cuDNN from 8.0 to 8.1 (backend-info)
- Added a `readonly` option to `drive.mount`
- Fixed bug where Xarray was not working (GitHub issue)
- Modified Markdown parsing to ignore block quote symbol within MathJax (GitHub issue)

2022-09-30

- Launched Pay As You Go, allowing premium GPU access without requiring a subscription
- Added vim and tcllib to our runtime image
- Fixed bug where open files were closed on kernel disconnect (GitHub issue)
- Fixed bug where the play button/execution indicator was not clickable when scrolled into the cell output (GitHub issue)
- Updated the styling for form titles so that they avoid obscuring the code editor
- Created a GitHub repo, backend-info, with the latest apt-list.txt and pip-freeze.txt files for the Colab runtime (GitHub issue)
- Added `files.upload_file(filename)` to upload a file from the browser to the runtime with a specified filename

2022-09-16

- Upgraded pymc from 3.11.0 to 4.1.4, jax from 0.3.14 to 0.3.17, jaxlib from 0.3.14 to 0.3.15, fsspec from 2022.8.1 to 2022.8.2
- Modified our save flow to avoid persisting Drive filenames as titles in notebook JSON
- Updated our Terms of Service
- Modified the `Jump to Cell` command to locate the cursor at the end of the command palette input (`Jump to cell` in Tools → Command palette in a notebook with section headings)
- Updated the styling of the Drive notebook comment UI

```
        k.layers.Conv2D(128, (3,3), activation='relu'),
        k.layers.BatchNormalization(),
        k.layers.MaxPooling2D(pool_size=(2)),
        k.layers.Dropout(0.3),
        k.layers.Conv2D(64, (3,3), padding='valid', activation='relu')
        k.layers.BatchNormalization(),
        k.layers.Flatten(),
        k.layers.Dense(64, activation='relu'),
        k.layers.Dense(len(index_label), activation='softmax'),

    ])
    print(model_4.summary())
    model_4_history = trainModel(model=model_4, epochs=50, optimizer='
```

```
Model: "sequential_3"
___________________________________________________________
 Layer (type)                Output Shape              Para
===========================================================
 conv2d (Conv2D)             (None, 124, 124, 256)     6656

 batch_normalization_6 (Bat  (None, 124, 124, 256)     1024
 chNormalization)

 max_pooling2d (MaxPooling2  (None, 62, 62, 256)       0
 D)

 dropout_6 (Dropout)         (None, 62, 62, 256)       0

 conv2d_1 (Conv2D)           (None, 60, 60, 128)       2950

 batch_normalization_7 (Bat  (None, 60, 60, 128)       512
 chNormalization)

 max_pooling2d_1 (MaxPoolin  (None, 30, 30, 128)       0
 g2D)

 dropout_7 (Dropout)         (None, 30, 30, 128)       0

 conv2d_2 (Conv2D)           (None, 28, 28, 64)        7379

 batch_normalization_8 (Bat  (None, 28, 28, 64)        256
 chNormalization)

 flatten_2 (Flatten)         (None, 50176)             0

 dense_7 (Dense)             (None, 64)                3211

 dense_8 (Dense)             (None, 3)                 195

===========================================================
Total params: 3588803 (13.69 MB)
Trainable params: 3587907 (13.69 MB)
Non-trainable params: 896 (3.50 KB)
___________________________________________________________
None

Epoch 1: val_loss improved from inf to 1.11297, saving mode

Epoch 2: val_loss did not improve from 1.11297

Epoch 3: val_loss did not improve from 1.11297

Epoch 4: val_loss did not improve from 1.11297

Epoch 5: val_loss did not improve from 1.11297

Epoch 6: val_loss did not improve from 1.11297

Epoch 7: val_loss did not improve from 1.11297
```

- Added support for terminating your runtime from code: `python from google.colab import runtime runtime.unassign()`
- Added regex filter support to the Recent notebooks dialog
- Inline google.colab.files.upload JS to fix `files.upload()` not working (GitHub issue)

2022-08-26

- Upgraded PyYAML from 3.13 to 6.0 (GitHub issue), drivefs from 61.0.3 to 62.0.1
- Upgraded TensorFlow from 2.8.2 to 2.9.1 and ipywidgets from 7.7.1 to 8.0.1 but rolled both back due to a number of user reports (GitHub issue, GitHub issue)
- Stop persisting inferred titles in notebook JSON (GitHub issue)
- Fix bug in background execution which affected some Pro+ users (GitHub issue)
- Fix bug where `Download as .py` incorrectly handled text cells ending in a double quote
- Fix bug for Pro and Pro+ users where we weren't honoring the preference (Tools → Settings) to use a temporary scratch notebook as the default landing page
- Provide undo/redo for scratch cells
- When writing ipynb files, serialize empty multiline strings as `[]` for better consistency with JupyterLab

2022-08-11

- Upgraded ipython from 5.5.0 to 7.9.0, fbprophet 0.7 to prophet 1.1, tensorflow-datasets from 4.0.1 to 4.6.0, drivefs from 60.0.2 to 61.0.3, pytorch from 1.12.0 to 1.12.1, numba from 0.51 to 0.56, and lxml from 4.2.0 to 4.9.1
- Loosened our `requests` version requirement (GitHub issue)
- Removed support for TensorFlow 1
- Added Help → Report Drive abuse for Drive notebooks
- Fixed indentation for Python lines ending in [
- Modified styling of tables in Markdown to left-align them rather than centering them
- Fixed special character replacement when copying interactive tables as Markdown
- Fixed ansi 8-bit color parsing (GitHub issue)
- Configured logging to preempt transitive imports and other loading from implicitly configuring the root logger
- Modified forms to use a value of `None` instead of causing a parse error when clearing raw and numeric-typed form fields

2022-07-22

- Update scipy from 1.4.1 to 1.7.3, drivefs from 59.0.3 to 60.0.2, pytorch from 1.11 to 1.12, jax & jaxlib from 0.3.8 to 0.3.14, opencv-python from 4.1.2.30 to 4.6.0.66,

```
plotHistory(model_4_history)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most
recent call last)
<ipython-input-1-2afade6330f1> in <cell line: 1>()
----> 1 plotHistory(model_4_history)

NameError: name 'plotHistory' is not defined
```

```
# evaluation
test_loss, test_acc  = k.models.load_model(bestModelPath).evaluate
print("The test Loss is :",test_loss)
print("The test Accuracy is :",test_acc*100)
```

```
inputShape = (X_train.shape[1], X_train.shape[2], 1)
model_5 = k.models.Sequential([
    k.layers.TimeDistributed(k.layers.Conv1D(256, 5), input_shape=
    k.layers.TimeDistributed(k.layers.BatchNormalization()),
    k.layers.TimeDistributed(k.layers.MaxPooling1D((2))),
    k.layers.TimeDistributed(k.layers.Dropout(0.3)),

    k.layers.TimeDistributed(k.layers.Conv1D(128, 3), input_shape=
    k.layers.TimeDistributed(k.layers.BatchNormalization()),
    k.layers.TimeDistributed(k.layers.MaxPooling1D((2))),
    k.layers.TimeDistributed(k.layers.Dropout(0.3)),
    k.layers.TimeDistributed(k.layers.Flatten())

], name="conv_3d7")

model_5.add(k.layers.Bidirectional(k.layers.LSTM(256, return_seque
model_5.add(k.layers.Dropout(0.3))

model_5.add(k.layers.Bidirectional(k.layers.LSTM(128)))
model_5.add(k.layers.Dropout(0.3))

model_5.add(k.layers.Dense(64, activation='relu'))
model_5.add(k.layers.Dropout(0.3))

model_5.add(k.layers.Dense(len(index_label), activation='softmax')

print(model_5.summary())
model_5_history = trainModel(model=model_5, epochs=100, optimizer=
```

```
plotHistory(model_5_history)
```

```
# evaluation
test_loss, test_acc  = k.models.load_model(bestModelPath).evaluate
print("The test Loss is :",test_loss)
print("The test Accuracy is :",test_acc*100)
```

Model using Image Data

```
modelPath = './best_model.hdf5'
ACCURACY_THRESHOLD = 0.95

class myCallback(k.callbacks.Callback):
    def on_epoch_end(self, epoch, logs={}):
```

spaCy from 3.3.1 to 3.4.0, and dlib from 19.18.0 to 19.24.0
- Fix `Open in tab` doc link which was rendering incorrectly (GitHub issue)
- Add a preference for the default tab orientation to the Site section of the settings menu under Tools → Settings
- Show a warning for `USE_AUTH_EPHEM` usage when running authenticate_user on a TPU runtime (code)

2022-07-01

- Add a preference for code font to the settings menu under Tools → Settings
- Update drivefs from 58.0.3 to 59.0.3 and spacy from 2.2.4 to 3.3.1
- Allow display_data and execute_result text outputs to wrap, matching behavior of JupyterLab (does not affect stream outputs/print statements).
- Improve LSP handling of some magics, esp. %%writefile (GitHub issue).
- Add a FAQ entry about the mount Drive button behavior and include link buttons for each FAQ entry.
- Fix bug where the notebook was sometimes hidden behind other tabs on load when in single pane view.
- Fix issue with inconsistent scrolling wher an editor is in multi-select mode.
- Fix bug where clicking on a link in a form would navigate away from the notebook
- Show a confirmation dialog before performing Replace all from the Find and replace pane.

2022-06-10

- Update drivefs from 57.0.5 to 58.0.3 and tensorflow from 2.8.0 to 2.8.2
- Support more than 100 repos in the GitHub repo selector shown in the open dialog and the clone to GitHub dialog
- Show full notebook names on hover in the open dialog
- Improve the color contrast for links, buttons, and the `ipywidgets.Accordion` widget in dark mode

2022-05-20

- Support URL params for linking to some common pref settings: force_theme=dark, force_corgi_mode=1, force_font_size=14. Params forced by URL are not persisted unless saved using Tools → Settings.
- Add a class `markdown-google-sans` to allow Markdown to render in Google Sans
- Update monaco-vim from 0.1.19 to 0.3.4
- Update drivefs from 55.0.3 to 57.0.5, jax from 0.3.4 to 0.3.8, and jaxlib from 0.3.2 to 0.3.7

2022-04-29

- Added 🦀 mode (under Miscellaneous in Tools → Settings)
- Added "Disconnect and delete runtime" option to the menu next to the Connect button

```
        if(logs.get('val_accuracy') > ACCURACY_THRESHOLD):
            print("\n\nStopping training as we have reached %2.2f%
            self.model.stop_training = True

acc_callback = myCallback()

cbs = [#k.callbacks.ReduceLROnPlateau(patience=3, verbose=1),
       k.callbacks.ModelCheckpoint(filepath=modelPath, monitor='va

def trainImgModel(model, epochs, optimizer, vb=1):
    batch_size = 64
    callback = myCallback()
    model.compile(optimizer=optimizer,
                  loss='sparse_categorical_crossentropy',
                  metrics='accuracy'
    )
    return model.fit(X_train_img, y_train,
                     validation_data=(X_test_img, y_test), epochs=
                     batch_size=batch_size, callbacks=cbs)

def plotHistory(history):
    print("Max. Validation Accuracy",max(history.history["val_accu
    pd.DataFrame(history.history).plot(figsize=(12,6))
    plt.show()


model_6 = k.models.Sequential([
    k.layers.Conv2D(256, 3, activation='relu', input_shape=(128, 1
    k.layers.BatchNormalization(),
    k.layers.MaxPooling2D(pool_size=(2)),
    k.layers.Dropout(0.2),
    k.layers.Conv2D(128, 3, activation='relu'),
    k.layers.BatchNormalization(),
    k.layers.MaxPooling2D(pool_size=(2)),
    k.layers.Dropout(0.2),
    k.layers.Conv2D(64, 3, padding='same', activation='relu'),
    k.layers.BatchNormalization(),
    k.layers.Flatten(),
    k.layers.Dense(64, activation='relu'),
    k.layers.Dense(len(index_label), activation='softmax'),

])
print(model_6.summary())
model_6_history = trainImgModel(model=model_6, epochs=100, optimiz


plotHistory(model_6_history)


# model evaluation
test_loss, test_acc  = k.models.load_model(bestModelPath).evaluate
print("The test Loss is :",test_loss)
print("The test Accuracy is :",test_acc*100)


# test_data = np.expand_dims(test_data, axis=3)


predictions = np.argmax(k.models.load_model(bestModelPath).predict
predictions


df_sub = pd.DataFrame({
    'Filename': test_files,
    'Class': list(map(lambda x:index_label[x], predictions))
})
df_sub.head()
```

- Improved rendering of filter options in an interactive table
- Added git-lfs to the base image
- Updated torch from 1.10.0 to 1.11.0, jupyter-core from 4.9.2 to 4.10.0, and cmake from 3.12.0 to 3.22.3
- Added more details to our FAQ about unsupported uses (using proxies, downloading torrents, etc.)
- Fixed issue with apt-get dependencies

2022-04-15

- Add an option in the file browser to show hidden files.
- Upgrade gdown from 4.2.0 to 4.4.0, google-api-core[grpc] from 1.26.0 to 1.31.5, and pytz from 2018.4 to 2022.1

2022-03-25

- Launched Pro/Pro+ to 12 additional countries: Australia, Bangladesh, Colombia, Hong Kong, Indonesia, Mexico New Zealand, Pakistan, Philippines, Singapore, Taiwan, and Vietnam
- Added `google.colab.auth.authenticate_se` to support using Service Account keys
- Update jax from 0.3.1 to 0.3.4 & jaxlib from 0.3.0 to 0.3.2
- Fixed an issue with Twitter previews of notebooks shared as GitHub Gists

2022-03-10

- Launched Pro/Pro+ to 10 new countries: Ireland, Israel, Italy, Morocco, the Netherlands, Poland, Spain, Switzerland, Turkey, and the United Arab Emirates
- Launched support for scheduling notebooks for Pro+ users
- Fixed bug in interactive datatables where filtering by number did not work
- Finished removing the python2 kernelspec

2022-02-25

- Made various accessibility improvements to the header
- Fix bug with forms run:auto where a form field change would trigger multiple runs
- Minor updates to the bigquery example notebook and snippet
- Include background execution setting in the sessions dialog for Pro+ users
- Update tensorflow-probability from 0.15 to 0.16
- Update jax from 0.2.25 to 0.3.1 & jaxlib from 0.1.71 to 0.3.0

2022-02-11

- Improve keyboard navigation for the open dialog
- Fix issue where nvidia-smi stopped reporting resource utilization for some users who were modifying the version of nvidia used
- Update tensorflow from 2.7 to 2.8, keras from 2.7 to 2.8, numpy from 1.19.5 to 1.21.5, tables from 3.4.4 to 3.7.0

```
submission_file = 'submission.csv'
df_sub.to_csv(submission_file, index=False)
```

2022-02-04

- Improve UX for opening content alongside your notebook, such as files opened from the file browser. This includes a multi-pane view and drag-drop support
- Better Twitter previews when sharing example Colab notebooks and notebooks opened from GitHub Gists
- Update pandas from 1.1.5 to 1.3.5
- Update openpyxl from 2.5.9 to 3.0.0 and pyarrow from 3.0.0 to 6.0.0
- Link to the release notes from the Help menu

2022-01-28

- Add a copy button to data tables
- Python LSP support for better completions and code diagnostics. This can be configured in the Editor Settings (Tools → Settings)
- Update gspread examples in our documentation
- Update gdown from 3.6 to 4.2

2022-01-21

- New documentation for the `google.colab` package
- Show GPU RAM in the resource usage tab
- Improved security for mounting Google Drive which disallows mounting Drive from accounts other than the one