Release notes ×

```
import pandas as pd
import seaborn as sns
import os
import numpy as np
import matplotlib.pyplot as plt
```

```
housing_df = pd.read_csv("/content/housing.csv")
```

```
# Use .info() to show the features (i.e. columns) in your dataset
housing_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20640 entries, 0 to 20639
Data columns (total 10 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   longitude           20640 non-null  float64
 1   latitude            20640 non-null  float64
 2   housing_median_age  20640 non-null  float64
 3   total_rooms         20640 non-null  float64
 4   total_bedrooms      20433 non-null  float64
 5   population          20640 non-null  float64
 6   households          20640 non-null  float64
 7   median_income       20640 non-null  float64
 8   median_house_value  20640 non-null  float64
 9   ocean_proximity     20640 non-null  object
dtypes: float64(9), object(1)
memory usage: 1.6+ MB
```

```
housing_df.shape
```

```
(20640, 10)
```

```
housing_df.head()
```

	longitude	latitude	housing_median_age	total_
0	-122.23	37.88	41.0	
1	-122.22	37.86	21.0	
2	-122.24	37.85	52.0	
3	-122.25	37.85	52.0	
4	-122.25	37.85	52.0	

Next steps: View recommended plots

```
housing_df.tail()
```

	longitude	latitude	housing_median_age	tc
20635	-121.09	39.48	25.0	
20636	-121.21	39.49	18.0	
20637	-121.22	39.43	17.0	
20638	-121.32	39.43	18.0	
20639	-121.24	39.37	16.0	

```
housing_df.describe()
```

	longitude	latitude	housing_median_a
count	20640.000000	20640.000000	20640.000
mean	-119.569704	35.631861	28.639
std	2.003532	2.135952	12.585
min	-124.350000	32.540000	1.000
25%	-121.800000	33.930000	18.000
50%	-118.490000	34.260000	29.000
75%	-118.010000	37.710000	37.000
max	-114.310000	41.950000	52.000

```
housing_df.isnull().sum()
```

```
longitude               0
latitude                0
housing_median_age      0
total_rooms             0
total_bedrooms        207
population              0
households              0
median_income           0
median_house_value      0
ocean_proximity         0
dtype: int64
```

```
# Calculate the % of missing data
housing_df['total_bedrooms'].isnull().sum()/housing_df.shape[0] *
```

```
1.002906976744186
```

```
from sklearn.impute import KNNImputer

# create a temporary copy of the dataset
housing_df_temp = housing_df.copy()

# retrieve columns with numerical data; will exclude the ocean_prc
columns_list = [col for col in housing_df_temp.columns if housing_

# extract columns that contain at least one missing value
new_column_list = [col for col in housing_df_temp.loc[:, housing_c

# update temp dataframe with numeric columns that have empty value
housing_df_temp = housing_df_temp[new_column_list]
```

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.

2024-02-21

- 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
  - 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_scr` Example notebook

```
# initialize KNNImputer to impute missing data using machine learr
knn = KNNImputer(n_neighbors = 3)

# fit function trains the model
knn.fit(housing_df_temp)

# transform the data using the model
# applies the transformation model (ie knn) to data
array_Values = knn.transform(housing_df_temp)

# convert the array values to a dataframe with the appropriate col
housing_df_temp = pd.DataFrame(array_Values, columns = new_column_


# confirm there are no columns with missing values
housing_df_temp.isnull().sum()
```

```
     total_bedrooms    0
     dtype: int64
```

```
# overlay the imputed column over the old column with missing valu

# loop through the list of columns and overlay each one
for column_name in new_column_list:
    housing_df[column_name] = housing_df_temp.replace(housing_df[c

# confirm columns no longer contain null data
housing_df.isnull().sum()
```

```
     longitude             0
     latitude              0
     housing_median_age    0
     total_rooms           0
     total_bedrooms        0
     population            0
     households            0
     median_income         0
     median_house_value    0
     ocean_proximity       0
     dtype: int64
```

```
# Plot the distribution of the target variable (median_house_value

# bins->amount of columns
plt.hist(housing_df['median_house_value'], bins=80)
plt.xlabel("House Values")



# We can see from the plot that the values of Median House Value a
# Most of the house are around 100,000-200,000 range
```

- 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 all 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

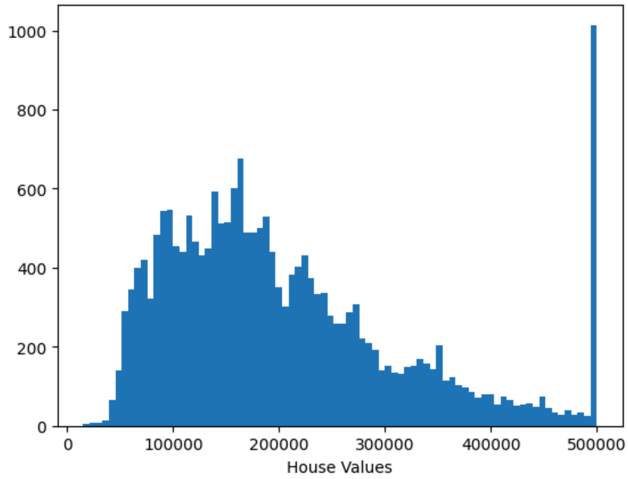
2023-11-27

- 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
  - 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

```
Text(0.5, 0, 'House Values')
```



```
# let's do histograms for the all the features to understand the c
# using housing_df as to not plot the encoded values for OCEAN_PRC
housing_df.hist(bins=50, figsize=(20,15))
```

- 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

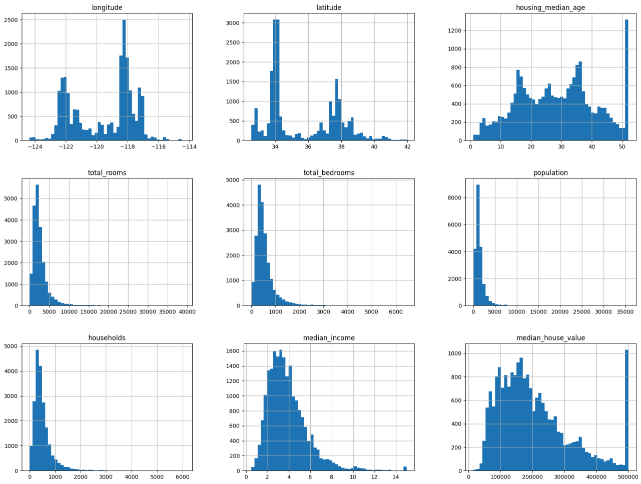
2023-09-22

- 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
  - 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

```
array([[<Axes: title={'center': 'longitude'}>,
        <Axes: title={'center': 'latitude'}>,
        <Axes: title={'center':
'housing_median_age'}>],
       [<Axes: title={'center': 'total_rooms'}>,
        <Axes: title={'center':
'total_bedrooms'}>,
        <Axes: title={'center': 'population'}>],
       [<Axes: title={'center': 'households'}>,
        <Axes: title={'center':
'median_income'}>,
        <Axes: title={'center':
'median_house_value'}>]], dtype=object)
```



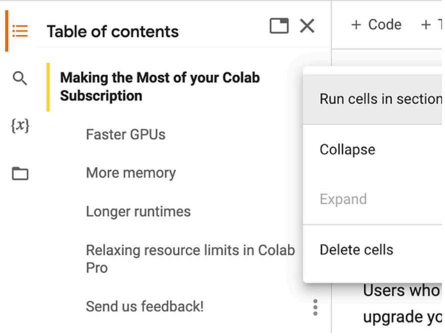
```
# Plot a graphical correlation matrix for each pair of columns in
corr = housing_df.corr() # data frame correlation function
print(corr)
```

```
                    longitude  latitude  housing_median_age  t
longitude            1.000000 -0.924664           -0.108197
latitude            -0.924664  1.000000            0.011173
housing_median_age  -0.108197  0.011173            1.000000
total_rooms          0.044568 -0.036100           -0.361262
total_bedrooms       0.069260 -0.066658           -0.318998
population           0.099773 -0.108785           -0.296244
households           0.055310 -0.071035           -0.302916
median_income       -0.015176 -0.079809           -0.119034
median_house_value  -0.045967 -0.144160            0.105623

                    total_bedrooms  population  households  me
longitude                 0.069260    0.099773    0.055310
latitude                 -0.066658   -0.108785   -0.071035
housing_median_age       -0.318998   -0.296244   -0.302916
total_rooms               0.927253    0.857126    0.918484
total_bedrooms            1.000000    0.873910    0.974725
population                0.873910    1.000000    0.907222
```

- 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)



Making the Most of your Colab Subscription

Faster GPUs

More memory

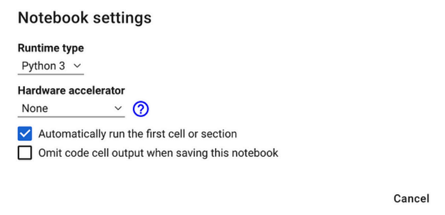
Longer runtimes

Relaxing resource limits in Colab Pro

Send us feedback!

Users who upgrade yc

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



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,

```
households                0.974725    0.907222    1.000000
median_income            -0.007682    0.004834    0.013033
median_house_value        0.049454   -0.024650    0.065843

                    median_house_value
longitude                    -0.045967
latitude                     -0.144160
housing_median_age            0.105623
total_rooms                   0.134153
total_bedrooms                0.049454
population                   -0.024650
households                    0.065843
median_income                 0.688075
median_house_value            1.000000
<ipython-input-24-3abd71ce2464>:2: FutureWarning: The default
  corr = housing_df.corr() # data frame correlation function
```

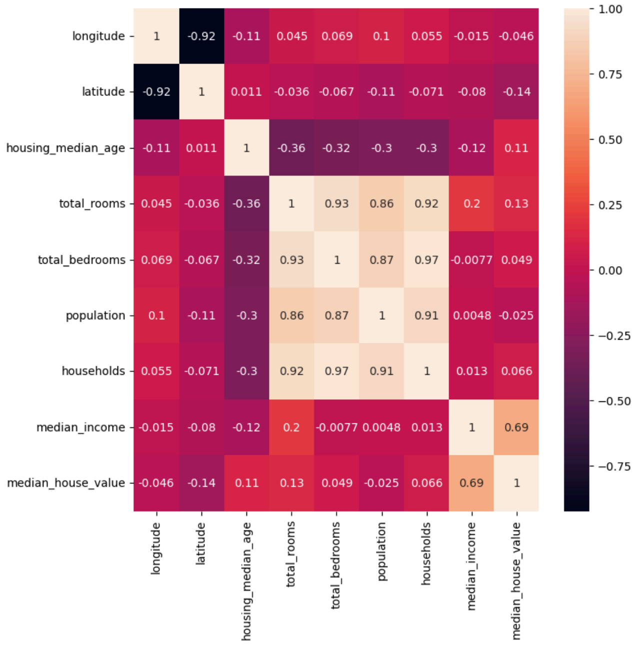
```
# make the heatmap larger in size
plt.figure(figsize = (8,8))

sns.heatmap(corr, annot=True)
plt.show()
```

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
  - 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



```
# Additionally we noted that several features (total_rooms,total_t
# so it's interesting to find out if a removal of a few of them wc

#  a new feature that is a ratio of the total rooms to households
housing_df['rooms_per_household'] = housing_df['total_rooms']/hous

# a new feature that is a ratio of the total bedrooms to the total
housing_df['bedrooms_per_room'] = housing_df['total_bedrooms']/hou

# a new feature that is a ratio of the population to the householc
housing_df['population_per_household']= housing_df['population']/h

# let's combine the latitude and longitude into 1
housing_df['coords'] = housing_df['longitude']/housing_df['latituc

housing_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20640 entries, 0 to 20639
Data columns (total 14 columns):
 #   Column                     Non-Null Count  Dtype
```

- 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
  - 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

```
 ---  ------                   --------------  -----
  0   longitude                20640 non-null  float64
  1   latitude                 20640 non-null  float64
  2   housing_median_age       20640 non-null  float64
  3   total_rooms              20640 non-null  float64
  4   total_bedrooms           20640 non-null  float64
  5   population               20640 non-null  float64
  6   households               20640 non-null  float64
  7   median_income            20640 non-null  float64
  8   median_house_value       20640 non-null  float64
  9   ocean_proximity          20640 non-null  object
  10  rooms_per_household      20640 non-null  float64
  11  bedrooms_per_room        20640 non-null  float64
  12  population_per_household 20640 non-null  float64
  13  coords                   20640 non-null  float64
 dtypes: float64(13), object(1)
 memory usage: 2.2+ MB
```

```
# remove total_rooms, households, total bedrooms, popluation, long
housing_df = housing_df.drop('total_rooms', axis=1)
housing_df = housing_df.drop('households', axis=1)
housing_df = housing_df.drop('total_bedrooms', axis=1)
housing_df = housing_df.drop('population', axis=1)
housing_df = housing_df.drop('longitude', axis=1)
housing_df = housing_df.drop('latitude', axis=1)

housing_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20640 entries, 0 to 20639
Data columns (total 8 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   housing_median_age        20640 non-null  float64
 1   median_income             20640 non-null  float64
 2   median_house_value        20640 non-null  float64
 3   ocean_proximity           20640 non-null  object
 4   rooms_per_household       20640 non-null  float64
 5   bedrooms_per_room         20640 non-null  float64
 6   population_per_household  20640 non-null  float64
 7   coords                    20640 non-null  float64
dtypes: float64(7), object(1)
memory usage: 1.3+ MB
```

```
#Heatmap after removing correlation

corr = housing_df.corr()

#make the heatmap larger in size
plt.figure(figsize = (7,7))

sns.heatmap(corr, annot=True)
plt.show()
```

- 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

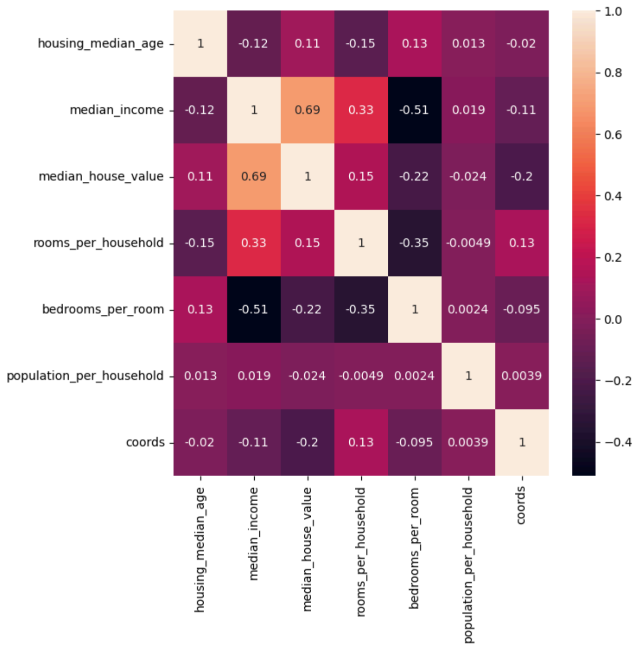
2023-03-31

- 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

- 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

```
<ipython-input-28-1264607259b1>:3: FutureWarning:
  corr = housing_df.corr()
```



```
#Encoding categorical data
# Most ML algorithms can only learn from numeric data (it's all Ma

# Let's review our data types again; showing that ocean_proximity
housing_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20640 entries, 0 to 20639
Data columns (total 8 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   housing_median_age        20640 non-null  float64
 1   median_income             20640 non-null  float64
 2   median_house_value        20640 non-null  float64
 3   ocean_proximity           20640 non-null  object
 4   rooms_per_household       20640 non-null  float64
 5   bedrooms_per_room         20640 non-null  float64
 6   population_per_household  20640 non-null  float64
 7   coords                    20640 non-null  float64
dtypes: float64(7), object(1)
memory usage: 1.3+ MB
```

- 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
    - 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

```
# let's see the unique categories for OCEAN_PROXIMITY
housing_df.ocean_proximity.unique()
```

```
array(['NEAR BAY', '<1H OCEAN', 'INLAND', 'NEAR OCEAN',
'ISLAND'],
      dtype=object)
```

```
# let's see the unique categories for OCEAN_PROXIMITY
housing_df.ocean_proximity.unique()
```

```
array(['NEAR BAY', '<1H OCEAN', 'INLAND', 'NEAR OCEAN',
'ISLAND'],
      dtype=object)
```

```
# Let's see how the Panda's get_dummies() function works (generate
print(pd.get_dummies(housing_df['ocean_proximity']))
```

```
       <1H OCEAN  INLAND  ISLAND  NEAR BAY  NEAR OCEAN
0              0       0       0         1           0
1              0       0       0         1           0
2              0       0       0         1           0
3              0       0       0         1           0
4              0       0       0         1           0
...          ...     ...     ...       ...         ...
20635          0       1       0         0           0
20636          0       1       0         0           0
20637          0       1       0         0           0
20638          0       1       0         0           0
20639          0       1       0         0           0

[20640 rows x 5 columns]
```

```
# let's replace the OCEAN_PROXIMITY column using get_dummies()
housing_df_encoded = pd.get_dummies(data=housing_df, columns=['oce

# print the first few observations; notice the old OCEAN_PROXIMITY
housing_df_encoded.head()
```

	housing_median_age	median_income	median_house
0	41.0	8.3252	4
1	21.0	8.3014	3
2	52.0	7.2574	3
3	52.0	5.6431	3
4	52.0	3.8462	3

Next steps: View recommended plots

  - 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
  - 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

```
#Train the model
import sklearn
from sklearn.model_selection import train_test_split

# remove spaces from column names and convert all to lowercase anc
housing_df_encoded.columns = [c.lower().replace(' ', '_').replace(

# Split target variable and feature variables
X = housing_df_encoded[['housing_median_age', 'median_income','bec
                        'ocean_proximity_inland','ocean_proximity_
y = housing_df_encoded['median_house_value']

print(X)
```

```
       housing_median_age  median_income  bedrooms_per_room  \
0                    41.0         8.3252           0.146591
1                    21.0         8.3014           0.155797
2                    52.0         7.2574           0.129516
3                    52.0         5.6431           0.184458
4                    52.0         3.8462           0.172096
...                   ...            ...                ...
20635                25.0         1.5603           0.224625
20636                18.0         2.5568           0.215208
20637                17.0         1.7000           0.215173
20638                18.0         1.8672           0.219892
20639                16.0         2.3886           0.221185

       population_per_household    coords  ocean_proximity__1h
0                      2.555556 -3.226769
1                      2.109842 -3.228209
2                      2.802260 -3.229590
3                      2.547945 -3.229855
4                      2.181467 -3.229855
...                         ...       ...
20635                  2.560606 -3.067123
20636                  3.122807 -3.069385
20637                  2.325635 -3.074309
20638                  2.123209 -3.076845
20639                  2.616981 -3.079502

       ocean_proximity_inland  ocean_proximity_island  \
0                           0                       0
1                           0                       0
2                           0                       0
3                           0                       0
4                           0                       0
...                       ...                     ...
20635                       1                       0
20636                       1                       0
20637                       1                       0
20638                       1                       0
20639                       1                       0

       ocean_proximity_near_bay  ocean_proximity_near_ocean
0                             1                           0
1                             1                           0
2                             1                           0
3                             1                           0
4                             1                           0
...                         ...                         ...
20635                         0                           0
20636                         0                           0
20637                         0                           0
20638                         0                           0
20639                         0                           0

[20640 rows x 10 columns]
```

    - 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
    - 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

```
# Split training & test data¶
# Splitting the data into training and testing sets in numpy array
# We train the model with 70% of the samples and test with the rem
# X -> array with the inputs; y -> array of the outputs
X_train, X_test, y_train, y_test = train_test_split(X, y, random_s

# Confirm how the data was split
print(X_train.shape)
print(X_test.shape)
print(y_train.shape)
print(y_test.shape)
```

```
(14448, 10)
(6192, 10)
(14448,)
(6192,)
```

```
#Linear Regression - Model Training¶
# Use scikit-learn's LinearRegression to train the model on both t
from sklearn.linear_model import LinearRegression

# Create a Linear regressor using all the feature variables
reg_model = LinearRegression()

# Train the model using the training sets
reg_model.fit(X_train, y_train)
```

```
▾ LinearRegression
LinearRegression()
```

```
#run the predictions on the training and testing data
y_pred_test = reg_model.predict(X_test)
```

```
#compare the actual values (ie, target) with the values predicted
pred_test_df = pd.DataFrame({'Actual': y_test, 'Predicted': y_pred

pred_test_df
```

	Actual	Predicted
20046	47700.0	103743.050896
3024	45800.0	92451.250932
15663	500001.0	219490.963844
20484	218600.0	283292.425471
9814	278000.0	244228.861575
...	...	...
17505	237500.0	210121.340663
13512	67300.0	74907.098235
10842	218400.0	216609.962950
16559	119400.0	127975.072923
5786	209800.0	202803.254310

6192 rows × 2 columns

Next steps: View recommended plots

- 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
- 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

```
# Determine accuracy uisng R^2
# R^2 : R squared is another way to evaluate the performance of a
# 1, means that the model is perfect and 0 means the the model wil
r2_reg_model_test = round(reg_model.score(X_test, y_test),2)

print("R^2 Test: {}".format(r2_reg_model_test))
```

```
    R^2 Test: 0.56
```

```
# try another machine learning algorithm : Randorm Forest
# Use scikit-learn’s Randorm Forest to train the model on both the
from sklearn.ensemble import RandomForestRegressor

# Create a  regressor using all the feature variables
rf_model = RandomForestRegressor(n_estimators=10,random_state=10)

# Train the model using the training sets
rf_model.fit(X_train, y_train)
```

```
▾                 RandomForestRegressor
RandomForestRegressor(n_estimators=10, random_stat
```

```
#run the predictions on the training and testing data
y_rf_pred_test = rf_model.predict(X_test)
```

```
#compare the actual values (ie, target) with the values predicted
rf_pred_test_df = pd.DataFrame({'Actual': y_test, 'Predicted': y_r

rf_pred_test_df
```

```
# Determine accuracy uisng R^2
from sklearn.metrics import r2_score, mean_squared_error

score = r2_score(y_test, y_rf_pred_test)

print("R^2 - {}%".format(round(score, 2) *100))
```

```
    R^2 - 75.0%
```

```
# Determine RMSE - Root Mean Squared Error on the test data
print('RMSE on test data: ',  mean_squared_error(y_test, y_rf_prec
```

```
    RMSE on test data:  57289.11495447338
```

```
# Determine feature importance - random forest algorithm is that i
# plot the 6 most important features
plt.figure(figsize=(10,6))
feat_importances = pd.Series(rf_model.feature_importances_, index
feat_importances.nlargest(6).plot(kind='barh');
```

- Inline google.colab.files.upload JS to fix `files.upload()` not working (GitHub issue)

2022-08-26

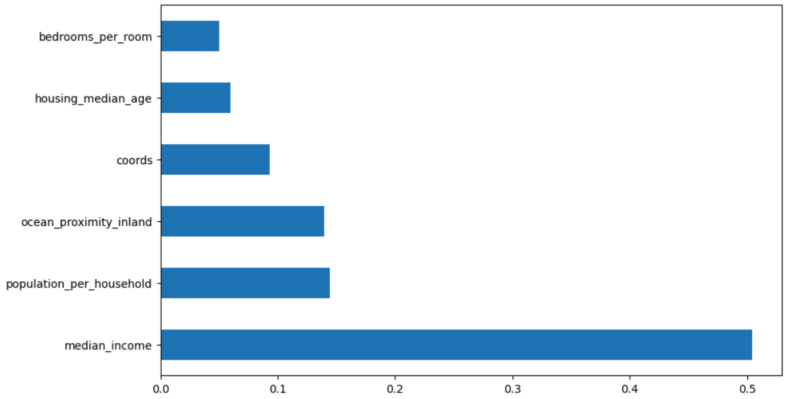
- Upgraded PyYAML from 3.13 to 6.0 (GitHub issue), drivefs from 61.0.3 to 62.0.1
- Upgraded TensorFlow from 2.8.2 to 2.9.1 and ipywidgets from 7.7.1 to 8.0.1 but rolled both back due to a number of user reports (GitHub issue, GitHub issue)
- Stop persisting inferred titles in notebook JSON (GitHub issue)
- 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, 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



```
# training data with 5 most important features
train_x_if = X_train[['bedrooms_per_room', 'housing_median_age', '
test_x_if = X_test[['bedrooms_per_room', 'housing_median_age', 'cc

# create an object of the RandfomForestRegressor Model
rf_model_if = RandomForestRegressor(n_estimators=10,random_state=1

# fit the model with the training data
rf_model_if.fit(train_x_if, y_train)

# predict the target on the test data
predict_test_with_if = rf_model_if.predict(test_x_if)


# Root Mean Squared Error on the train and test data
print('RMSE on test data: ',  mean_squared_error(y_test, predict_t
```

```
     RMSE on test data:  57366.910692045196
```

```
pip install xgboost
```

```
     Requirement already satisfied: xgboost in /usr/local/lib/pytho
     Requirement already satisfied: numpy in /usr/local/lib/python3
     Requirement already satisfied: scipy in /usr/local/lib/python3
```

```
# Extreme Gradient Boosting (XGBoost) is an open-source library th
# Use the scikit-learn wrapper classes: XGBRegressor and XGBClassi

# try another machine learning algorithm : XGBoost
from xgboost import XGBRegressor

xgb_model = XGBRegressor()


# Train the model using the training sets
xgb_model.fit(X_train, y_train)
```

- 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 when 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
- 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

```
XGBRegressor
XGBRegressor(base_score=None, booster=None, callbacks=None,
             colsample_bylevel=None, colsample_bynode=None,
             colsample_bytree=None, device=None, early_stoppi
             enable_categorical=False, eval_metric=None, feat
             gamma=None, grow_policy=None, importance_type=No
             interaction_constraints=None, learning_rate=None
             max_cat_threshold=None, max_cat_to_onehot=None,
             max_delta_step=None, max_depth=None, max_leaves=
             min_child_weight=None, missing=nan, monotone_con
             multi_strategy=None, n_estimators=None, n_jobs=N
             num_parallel_tree=None, random_state=None, ...)
```

```
#run the predictions on the training and testing data
y_xgb_pred_test = xgb_model.predict(X_test)


#compare the actual values (ie, target) with the values predicted
xgb_pred_test_df = pd.DataFrame({'Actual': y_test, 'Predicted': y_

xgb_pred_test_df
```

	Actual	Predicted
20046	47700.0	66404.914062
3024	45800.0	86681.765625
15663	500001.0	449666.093750
20484	218600.0	262887.281250
9814	278000.0	218322.796875
...	...	...
17505	237500.0	227466.500000
13512	67300.0	64712.433594
10842	218400.0	218226.109375
16559	119400.0	123181.968750
5786	209800.0	227016.828125

6192 rows × 2 columns

Next steps: View recommended plots

```
fig= plt.figure(figsize=(8,8))
xgb_pred_test_df = xgb_pred_test_df.reset_index()
xgb_pred_test_df = xgb_pred_test_df.drop(['index'],axis=1)
plt.plot(xgb_pred_test_df[:50])
plt.legend(['Actual value','Predicted value'])
```

- 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_serv` 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

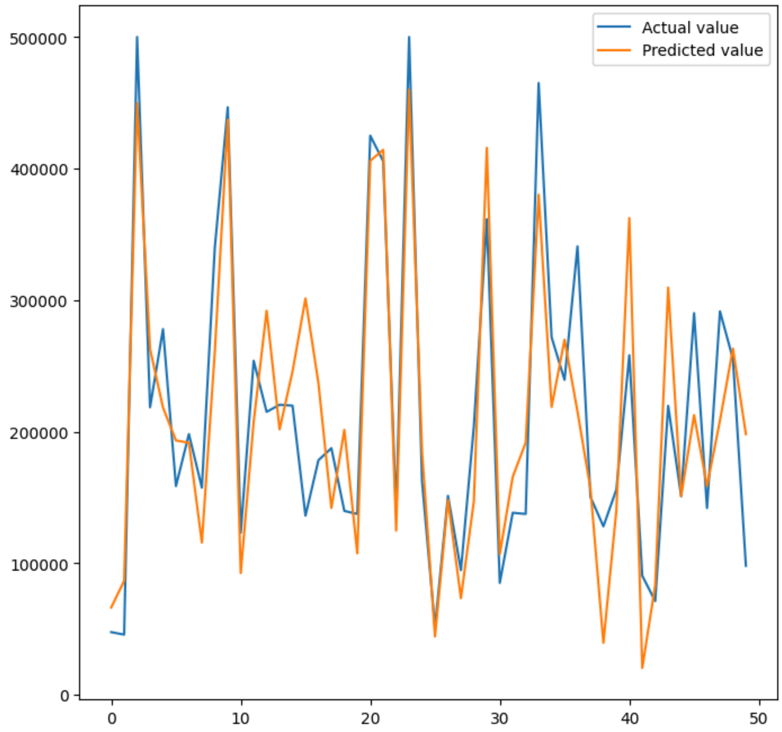
2022-02-11

- 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

2022-02-04

- Improve UX for opening content alongside your notebook, such as files opened from the file browser. This

```
<matplotlib.legend.Legend at 0x7dc9368b9510>
```



```
from sklearn.metrics import r2_score

score = r2_score(y_test, y_xgb_pred_test)

print("R^2 - {}%".format(round(score, 2) *100))
```

```
     R^2 - 78.0%
```

```
# Determine mean square error and root mean square error
from sklearn.metrics import mean_squared_error
import math

mse = mean_squared_error(y_test, y_xgb_pred_test)
rmse = math.sqrt(mean_squared_error(y_test, y_xgb_pred_test))

print(mse)
print(rmse)
```

```
     2939759040.9080276
     54219.5448238735
```

```
# Calculate mean absolute error(any large error)
from sklearn.metrics import mean_absolute_error

print(mean_absolute_error(y_test, y_xgb_pred_test))
```

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 currently executing the notebook

2022-01-14

- Add a preference (Tools → Settings) to use a temporary scratch notebook as the default landing page
- Fix bug where `/` and `:` weren't working in VIM mode
- Update gspread from 3.0 to 3.4
- Update the Colab Marketplace VM image

```
36285.050324826894
```

```
# We can build and score a model on multiple folds using cross-vali
from sklearn.model_selection import RepeatedKFold
from sklearn.model_selection import cross_val_score


# define model evaluation method
cv = RepeatedKFold(n_splits=10, n_repeats=3, random_state=1)

scores = cross_val_score(xgb_model, X, y, scoring='r2', error_score

#average of all the r2 scores across runs
print(scores.mean())
```

```
[Parallel(n_jobs=-1)]: Using backend LokyBackend with 2 concur
0.7850403811484551
[Parallel(n_jobs=-1)]: Done  30 out of  30 | elapsed:   11.9s
```

```
# determine hyperparameter available for tuning
xgb_model.get_params()
```

```
{'objective': 'reg:squarederror',
 'base_score': None,
 'booster': None,
 'callbacks': None,
```