svm

December 12, 2024

```python
# This mounts your Google Drive to the Colab VM.
from google.colab import drive
drive.mount('/content/drive')

# TODO: Enter the foldername in your Drive where you have saved the unzipped
# assignment folder, e.g. 'cs6353/assignments/assignment2/'
FOLDERNAME = 'cs6353/assignments/assignment2/'
assert FOLDERNAME is not None, "[!] Enter the foldername."

# Now that we've mounted your Drive, this ensures that
# the Python interpreter of the Colab VM can load
# python files from within it.
import sys
sys.path.append('/content/drive/My Drive/{}'.format(FOLDERNAME))

# This downloads the CIFAR-10 dataset to your Drive
# if it doesn't already exist.
%cd /content/drive/My\ Drive/$FOLDERNAME/cs6353/datasets/
!bash get_datasets.sh
%cd /content/drive/My\ Drive/$FOLDERNAME

# Install requirements from colab_requirements.txt
# TODO: Please change your path below to the colab_requirements.txt file
! python -m pip install -r /content/drive/My\ Drive/$FOLDERNAME/
  colab_requirements.txt
```

```
Mounted at /content/drive
/content/drive/My Drive/cs6353/assignments/assignment2/cs6353/datasets
--2024-12-06 01:10:00--  http://www.cs.toronto.edu/~kriz/cifar-10-python.tar.gz
Resolving www.cs.toronto.edu (www.cs.toronto.edu)… 128.100.3.30
Connecting to www.cs.toronto.edu (www.cs.toronto.edu)|128.100.3.30|:80…
connected.
HTTP request sent, awaiting response… 200 OK
Length: 170498071 (163M) [application/x-gzip]
Saving to: 'cifar-10-python.tar.gz'

cifar-10-python.tar 100%[===================>] 162.60M  51.0MB/s    in 3.4s
```

```
2024-12-06 01:10:04 (47.4 MB/s) - 'cifar-10-python.tar.gz' saved
[170498071/170498071]

cifar-10-batches-py/
cifar-10-batches-py/data_batch_4
cifar-10-batches-py/readme.html
cifar-10-batches-py/test_batch
cifar-10-batches-py/data_batch_3
cifar-10-batches-py/batches.meta
cifar-10-batches-py/data_batch_2
cifar-10-batches-py/data_batch_5
cifar-10-batches-py/data_batch_1
/content/drive/My Drive/cs6353/assignments/assignment2
Requirement already satisfied: anyio==3.7.1 in /usr/local/lib/python3.10/dist-
packages (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 1)) (3.7.1)
Collecting appnope==0.1.3 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 2))
  Downloading appnope-0.1.3-py2.py3-none-any.whl.metadata (1.2 kB)
Requirement already satisfied: argon2-cffi==23.1.0 in
/usr/local/lib/python3.10/dist-packages (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 3)) (23.1.0)
Requirement already satisfied: argon2-cffi-bindings==21.2.0 in
/usr/local/lib/python3.10/dist-packages (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 4)) (21.2.0)
Collecting arrow==1.2.3 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 5))
  Downloading arrow-1.2.3-py3-none-any.whl.metadata (6.9 kB)
Collecting asttokens==2.2.1 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 6))
  Downloading asttokens-2.2.1-py2.py3-none-any.whl.metadata (4.8 kB)
Collecting async-lru==2.0.4 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 7))
  Downloading async_lru-2.0.4-py3-none-any.whl.metadata (4.5 kB)
Collecting attrs==23.1.0 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 8))
  Downloading attrs-23.1.0-py3-none-any.whl.metadata (11 kB)
Collecting Babel==2.12.1 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 9))
  Downloading Babel-2.12.1-py3-none-any.whl.metadata (1.3 kB)
Requirement already satisfied: backcall==0.2.0 in
/usr/local/lib/python3.10/dist-packages (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 10)) (0.2.0)
Collecting beautifulsoup4==4.12.2 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 11))
  Downloading beautifulsoup4-4.12.2-py3-none-any.whl.metadata (3.6 kB)
Collecting bleach==6.0.0 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 12))
```

```
  Downloading bleach-6.0.0-py3-none-any.whl.metadata (29 kB)
Collecting certifi==2023.7.22 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 13))
  Downloading certifi-2023.7.22-py3-none-any.whl.metadata (2.2 kB)
Collecting cffi==1.15.1 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 14))
  Downloading
cffi-1.15.1-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata
(1.1 kB)
Collecting charset-normalizer==3.2.0 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 15))
  Downloading charset_normalizer-3.2.0-cp310-cp310-manylinux_2_17_x86_64.manylin
ux2014_x86_64.whl.metadata (31 kB)
Collecting comm==0.1.4 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 16))
  Downloading comm-0.1.4-py3-none-any.whl.metadata (4.2 kB)
Collecting contourpy==1.1.0 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 17))
  Downloading contourpy-1.1.0-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x8
6_64.whl.metadata (5.7 kB)
Collecting cycler==0.11.0 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 18))
  Downloading cycler-0.11.0-py3-none-any.whl.metadata (785 bytes)
Collecting debugpy==1.6.7.post1 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 19))
  Downloading debugpy-1.6.7.post1-cp310-cp310-manylinux_2_17_x86_64.manylinux201
4_x86_64.whl.metadata (1.1 kB)
Requirement already satisfied: decorator<=5.0 in /usr/local/lib/python3.10/dist-
packages (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 20)) (4.4.2)
Requirement already satisfied: defusedxml==0.7.1 in
/usr/local/lib/python3.10/dist-packages (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 21)) (0.7.1)
Collecting executing==1.2.0 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 22))
  Downloading executing-1.2.0-py2.py3-none-any.whl.metadata (8.9 kB)
Collecting fastjsonschema==2.18.0 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 23))
  Downloading fastjsonschema-2.18.0-py3-none-any.whl.metadata (2.0 kB)
Collecting fonttools==4.42.1 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 24))
  Downloading fonttools-4.42.1-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x
86_64.whl.metadata (150 kB)
                              151.0/151.0

kB 6.5 MB/s eta 0:00:00
Collecting fqdn==1.5.1 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 25))
```

```
    Downloading fqdn-1.5.1-py3-none-any.whl.metadata (1.4 kB)
Collecting idna==3.4 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 26))
  Downloading idna-3.4-py3-none-any.whl.metadata (9.8 kB)
Collecting imageio==2.31.1 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 27))
  Downloading imageio-2.31.1-py3-none-any.whl.metadata (4.7 kB)
Requirement already satisfied: ipykernel<=5.5.6 in
/usr/local/lib/python3.10/dist-packages (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 28)) (5.5.6)
Requirement already satisfied: ipython<=7.34.0 in
/usr/local/lib/python3.10/dist-packages (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 29)) (7.34.0)
Collecting isoduration==20.11.0 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 30))
  Downloading isoduration-20.11.0-py3-none-any.whl.metadata (5.7 kB)
Collecting jedi==0.19.0 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 31))
  Downloading jedi-0.19.0-py2.py3-none-any.whl.metadata (22 kB)
Collecting Jinja2==3.1.2 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 32))
  Downloading Jinja2-3.1.2-py3-none-any.whl.metadata (3.5 kB)
Collecting json5==0.9.14 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 33))
  Downloading json5-0.9.14-py2.py3-none-any.whl.metadata (10 kB)
Collecting jsonpointer==2.4 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 34))
  Downloading jsonpointer-2.4-py2.py3-none-any.whl.metadata (2.5 kB)
Collecting jsonschema==4.19.0 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 35))
  Downloading jsonschema-4.19.0-py3-none-any.whl.metadata (8.2 kB)
Collecting jsonschema-specifications==2023.7.1 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 36))
  Downloading jsonschema_specifications-2023.7.1-py3-none-any.whl.metadata (2.8
kB)
Collecting jupyter-events==0.7.0 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 37))
  Downloading jupyter_events-0.7.0-py3-none-any.whl.metadata (5.5 kB)
Collecting jupyter-lsp==2.2.0 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 38))
  Downloading jupyter_lsp-2.2.0-py3-none-any.whl.metadata (1.8 kB)
Requirement already satisfied: jupyter_client<8.0 in
/usr/local/lib/python3.10/dist-packages (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 39)) (6.1.12)
Collecting jupyter_core==5.3.1 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 40))
  Downloading jupyter_core-5.3.1-py3-none-any.whl.metadata (3.4 kB)
Collecting jupyter_server==2.7.2 (from -r /content/drive/My
```

Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 41))
  Downloading jupyter_server-2.7.2-py3-none-any.whl.metadata (8.6 kB)
Collecting jupyter_server_terminals==0.4.4 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 42))
  Downloading jupyter_server_terminals-0.4.4-py3-none-any.whl.metadata (6.3 kB)
Collecting jupyterlab==4.0.5 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 43))
  Downloading jupyterlab-4.0.5-py3-none-any.whl.metadata (15 kB)
Collecting jupyterlab-pygments==0.2.2 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 44))
  Downloading jupyterlab_pygments-0.2.2-py2.py3-none-any.whl.metadata (1.9 kB)
Collecting jupyterlab_server==2.24.0 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 45))
  Downloading jupyterlab_server-2.24.0-py3-none-any.whl.metadata (5.8 kB)
Collecting kiwisolver==1.4.5 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 46))
  Downloading kiwisolver-1.4.5-cp310-cp310-manylinux_2_12_x86_64.manylinux2010_x
86_64.whl.metadata (6.4 kB)
Collecting MarkupSafe==2.1.3 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 47))
  Downloading MarkupSafe-2.1.3-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x
86_64.whl.metadata (3.0 kB)
Collecting matplotlib==3.7.2 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 48))
  Downloading matplotlib-3.7.2-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x
86_64.whl.metadata (5.6 kB)
Collecting matplotlib-inline==0.1.6 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 49))
  Downloading matplotlib_inline-0.1.6-py3-none-any.whl.metadata (2.8 kB)
Collecting mistune==3.0.1 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 50))
  Downloading mistune-3.0.1-py3-none-any.whl.metadata (1.7 kB)
Collecting nbclient==0.8.0 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 51))
  Downloading nbclient-0.8.0-py3-none-any.whl.metadata (7.8 kB)
Collecting nbconvert==7.7.4 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 52))
  Downloading nbconvert-7.7.4-py3-none-any.whl.metadata (8.0 kB)
Collecting nbformat==5.9.2 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 53))
  Downloading nbformat-5.9.2-py3-none-any.whl.metadata (3.4 kB)
Collecting nest-asyncio==1.5.7 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 54))
  Downloading nest_asyncio-1.5.7-py3-none-any.whl.metadata (2.7 kB)
Collecting notebook_shim==0.2.3 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 55))
  Downloading notebook_shim-0.2.3-py3-none-any.whl.metadata (4.0 kB)
Collecting numpy<1.24,>=1.22 (from -r /content/drive/My

Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 56))
  Downloading
numpy-1.23.5-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata
(2.3 kB)
Collecting overrides==7.4.0 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 57))
  Downloading overrides-7.4.0-py3-none-any.whl.metadata (5.7 kB)
Collecting packaging==23.1 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 58))
  Downloading packaging-23.1-py3-none-any.whl.metadata (3.1 kB)
Collecting pandas<=1.5.3 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 59))
  Downloading
pandas-1.5.3-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata
(11 kB)
Collecting pandocfilters==1.5.0 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 60))
  Downloading pandocfilters-1.5.0-py2.py3-none-any.whl.metadata (9.0 kB)
Collecting parso==0.8.3 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 61))
  Downloading parso-0.8.3-py2.py3-none-any.whl.metadata (7.5 kB)
Collecting pexpect==4.8.0 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 62))
  Downloading pexpect-4.8.0-py2.py3-none-any.whl.metadata (2.2 kB)
Requirement already satisfied: pickleshare==0.7.5 in
/usr/local/lib/python3.10/dist-packages (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 63)) (0.7.5)
Collecting Pillow==10.0.0 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 64))
  Downloading Pillow-10.0.0-cp310-cp310-manylinux_2_28_x86_64.whl.metadata (9.5
kB)
Collecting platformdirs==3.10.0 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 65))
  Downloading platformdirs-3.10.0-py3-none-any.whl.metadata (11 kB)
Collecting prometheus-client==0.17.1 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 66))
  Downloading prometheus_client-0.17.1-py3-none-any.whl.metadata (24 kB)
Collecting prompt-toolkit==3.0.39 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 67))
  Downloading prompt_toolkit-3.0.39-py3-none-any.whl.metadata (6.4 kB)
Requirement already satisfied: psutil==5.9.5 in /usr/local/lib/python3.10/dist-
packages (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 68)) (5.9.5)
Requirement already satisfied: ptyprocess==0.7.0 in
/usr/local/lib/python3.10/dist-packages (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 69)) (0.7.0)
Collecting pure-eval==0.2.2 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 70))

```
  Downloading pure_eval-0.2.2-py3-none-any.whl.metadata (6.2 kB)
Collecting pycparser==2.21 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 71))
  Downloading pycparser-2.21-py2.py3-none-any.whl.metadata (1.1 kB)
Collecting Pygments==2.16.1 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 72))
  Downloading Pygments-2.16.1-py3-none-any.whl.metadata (2.5 kB)
Collecting pyparsing==3.0.9 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 73))
  Downloading pyparsing-3.0.9-py3-none-any.whl.metadata (4.2 kB)
Requirement already satisfied: python-dateutil==2.8.2 in
/usr/local/lib/python3.10/dist-packages (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 74)) (2.8.2)
Collecting python-json-logger==2.0.7 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 75))
  Downloading python_json_logger-2.0.7-py3-none-any.whl.metadata (6.5 kB)
Collecting pytz==2023.3 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 76))
  Downloading pytz-2023.3-py2.py3-none-any.whl.metadata (22 kB)
Collecting PyYAML==6.0.1 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 77))
  Downloading
PyYAML-6.0.1-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata
(2.1 kB)
Requirement already satisfied: pyzmq<25 in /usr/local/lib/python3.10/dist-
packages (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 78)) (24.0.1)
Collecting referencing==0.30.2 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 79))
  Downloading referencing-0.30.2-py3-none-any.whl.metadata (2.6 kB)
WARNING: Retrying (Retry(total=4, connect=None, read=None, redirect=None,
status=None)) after connection broken by 'ProtocolError('Connection aborted.',
RemoteDisconnected('Remote end closed connection without response'))':
/simple/requests/
Collecting requests==2.31.0 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 80))
  Downloading requests-2.31.0-py3-none-any.whl.metadata (4.6 kB)
Collecting rfc3339-validator==0.1.4 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 81))
  Downloading rfc3339_validator-0.1.4-py2.py3-none-any.whl.metadata (1.5 kB)
Collecting rfc3986-validator==0.1.1 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 82))
  Downloading rfc3986_validator-0.1.1-py2.py3-none-any.whl.metadata (1.7 kB)
Collecting rpds-py==0.9.2 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 83))
```

```
  Downloading rpds_py-0.9.2-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_
64.whl.metadata (3.7 kB)
Collecting scipy==1.11.2 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 84))
  Downloading
scipy-1.11.2-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata
(59 kB)
                            59.1/59.1 kB
4.1 MB/s eta 0:00:00
Collecting seaborn==0.12.2 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 85))
  Downloading seaborn-0.12.2-py3-none-any.whl.metadata (5.4 kB)
WARNING: Retrying (Retry(total=4, connect=None, read=None, redirect=None,
status=None)) after connection broken by 'ProtocolError('Connection aborted.',
RemoteDisconnected('Remote end closed connection without response'))':
/simple/send2trash/
Collecting Send2Trash==1.8.2 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 86))
  Downloading Send2Trash-1.8.2-py3-none-any.whl.metadata (4.0 kB)
Requirement already satisfied: six==1.16.0 in /usr/local/lib/python3.10/dist-
packages (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 87)) (1.16.0)
Collecting sniffio==1.3.0 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 88))
  Downloading sniffio-1.3.0-py3-none-any.whl.metadata (3.6 kB)
Collecting soupsieve==2.4.1 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 89))
  Downloading soupsieve-2.4.1-py3-none-any.whl.metadata (4.7 kB)
Collecting stack-data==0.6.2 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 90))
  Downloading stack_data-0.6.2-py3-none-any.whl.metadata (18 kB)
Collecting terminado==0.17.1 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 91))
  Downloading terminado-0.17.1-py3-none-any.whl.metadata (5.9 kB)
Collecting tinycss2==1.2.1 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 92))
  Downloading tinycss2-1.2.1-py3-none-any.whl.metadata (3.0 kB)
Collecting tornado<=6.3.2 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 93))
  Downloading tornado-6.3.2-cp38-abi3-manylinux_2_5_x86_64.manylinux1_x86_64.man
ylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (2.5 kB)
Collecting traitlets==5.9.0 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 94))
  Downloading traitlets-5.9.0-py3-none-any.whl.metadata (10 kB)
Collecting tzdata==2023.3 (from -r /content/drive/My
```

Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 95))
  Downloading tzdata-2023.3-py2.py3-none-any.whl.metadata (1.4 kB)
Collecting uri-template==1.3.0 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 96))
  Downloading uri_template-1.3.0-py3-none-any.whl.metadata (8.8 kB)
Collecting urllib3==2.0.4 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 97))
  Downloading urllib3-2.0.4-py3-none-any.whl.metadata (6.6 kB)
Collecting wcwidth==0.2.6 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 98))
  Downloading wcwidth-0.2.6-py2.py3-none-any.whl.metadata (11 kB)
Collecting webcolors==1.13 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 99))
  Downloading webcolors-1.13-py3-none-any.whl.metadata (2.6 kB)
Requirement already satisfied: webencodings==0.5.1 in
/usr/local/lib/python3.10/dist-packages (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 100)) (0.5.1)
Collecting websocket-client==1.6.2 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 101))
  Downloading websocket_client-1.6.2-py3-none-any.whl.metadata (7.5 kB)
Requirement already satisfied: exceptiongroup in /usr/local/lib/python3.10/dist-
packages (from anyio==3.7.1->-r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 1)) (1.2.2)
Requirement already satisfied: typing-extensions>=4.0.0 in
/usr/local/lib/python3.10/dist-packages (from async-lru==2.0.4->-r
/content/drive/My Drive/cs6353/assignments/assignment2//colab_requirements.txt
(line 7)) (4.12.2)
Collecting jupyter_client<8.0 (from -r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 39))
  Downloading jupyter_client-7.4.9-py3-none-any.whl.metadata (8.5 kB)
Requirement already satisfied: tomli in /usr/local/lib/python3.10/dist-packages
(from jupyterlab==4.0.5->-r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 43)) (2.2.1)
Requirement already satisfied: ipython-genutils in
/usr/local/lib/python3.10/dist-packages (from ipykernel<=5.5.6->-r
/content/drive/My Drive/cs6353/assignments/assignment2//colab_requirements.txt
(line 28)) (0.2.0)
Requirement already satisfied: setuptools>=18.5 in
/usr/local/lib/python3.10/dist-packages (from ipython<=7.34.0->-r
/content/drive/My Drive/cs6353/assignments/assignment2//colab_requirements.txt
(line 29)) (75.1.0)
Requirement already satisfied: entrypoints in /usr/local/lib/python3.10/dist-
packages (from jupyter_client<8.0->-r /content/drive/My
Drive/cs6353/assignments/assignment2//colab_requirements.txt (line 39)) (0.4)
Downloading appnope-0.1.3-py2.py3-none-any.whl (4.4 kB)
Downloading arrow-1.2.3-py3-none-any.whl (66 kB)
                         66.4/66.4 kB
4.1 MB/s eta 0:00:00

```
Downloading asttokens-2.2.1-py2.py3-none-any.whl (26 kB)
Downloading async_lru-2.0.4-py3-none-any.whl (6.1 kB)
Downloading attrs-23.1.0-py3-none-any.whl (61 kB)
                         61.2/61.2 kB
3.9 MB/s eta 0:00:00
Downloading Babel-2.12.1-py3-none-any.whl (10.1 MB)
                         10.1/10.1 MB
60.3 MB/s eta 0:00:00
Downloading beautifulsoup4-4.12.2-py3-none-any.whl (142 kB)
                         143.0/143.0 kB
6.9 MB/s eta 0:00:00
Downloading bleach-6.0.0-py3-none-any.whl (162 kB)
                         162.5/162.5 kB
10.3 MB/s eta 0:00:00
Downloading certifi-2023.7.22-py3-none-any.whl (158 kB)
                         158.3/158.3 kB
10.2 MB/s eta 0:00:00
Downloading
cffi-1.15.1-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (441 kB)
                         441.8/441.8 kB
23.2 MB/s eta 0:00:00
Downloading charset_normalizer-3.2.0-cp310-cp310-manylinux_2_17_x86_64.man
ylinux2014_x86_64.whl (201 kB)
                         201.8/201.8 kB
13.3 MB/s eta 0:00:00
Downloading comm-0.1.4-py3-none-any.whl (6.6 kB)
Downloading
contourpy-1.1.0-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (300
kB)
                         300.7/300.7 kB
18.7 MB/s eta 0:00:00
Downloading cycler-0.11.0-py3-none-any.whl (6.4 kB)
Downloading
debugpy-1.6.7.post1-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl
(3.0 MB)
                         3.0/3.0 MB
67.1 MB/s eta 0:00:00
Downloading executing-1.2.0-py2.py3-none-any.whl (24 kB)
Downloading fastjsonschema-2.18.0-py3-none-any.whl (23 kB)
Downloading
fonttools-4.42.1-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (4.5
MB)
                         4.5/4.5 MB
83.6 MB/s eta 0:00:00
Downloading fqdn-1.5.1-py3-none-any.whl (9.1 kB)
Downloading idna-3.4-py3-none-any.whl (61 kB)
                         61.5/61.5 kB
4.3 MB/s eta 0:00:00
```

```
Downloading imageio-2.31.1-py3-none-any.whl (313 kB)
                           313.2/313.2 kB
20.3 MB/s eta 0:00:00
Downloading isoduration-20.11.0-py3-none-any.whl (11 kB)
Downloading jedi-0.19.0-py2.py3-none-any.whl (1.6 MB)
                           1.6/1.6 MB
56.9 MB/s eta 0:00:00
Downloading Jinja2-3.1.2-py3-none-any.whl (133 kB)
                           133.1/133.1 kB
9.4 MB/s eta 0:00:00
Downloading json5-0.9.14-py2.py3-none-any.whl (19 kB)
Downloading jsonpointer-2.4-py2.py3-none-any.whl (7.8 kB)
Downloading jsonschema-4.19.0-py3-none-any.whl (83 kB)
                           83.4/83.4 kB
6.0 MB/s eta 0:00:00
Downloading jsonschema_specifications-2023.7.1-py3-none-any.whl (17 kB)
Downloading jupyter_events-0.7.0-py3-none-any.whl (18 kB)
Downloading jupyter_lsp-2.2.0-py3-none-any.whl (65 kB)
                           66.0/66.0 kB
4.7 MB/s eta 0:00:00
Downloading jupyter_core-5.3.1-py3-none-any.whl (93 kB)
                           93.7/93.7 kB
7.0 MB/s eta 0:00:00
Downloading jupyter_server-2.7.2-py3-none-any.whl (375 kB)
                           375.3/375.3 kB
25.9 MB/s eta 0:00:00
Downloading jupyter_server_terminals-0.4.4-py3-none-any.whl (13 kB)
Downloading jupyterlab-4.0.5-py3-none-any.whl (9.2 MB)
                           9.2/9.2 MB
89.6 MB/s eta 0:00:00
Downloading jupyterlab_pygments-0.2.2-py2.py3-none-any.whl (21 kB)
Downloading jupyterlab_server-2.24.0-py3-none-any.whl (57 kB)
                           57.3/57.3 kB
4.2 MB/s eta 0:00:00
Downloading
kiwisolver-1.4.5-cp310-cp310-manylinux_2_12_x86_64.manylinux2010_x86_64.whl (1.6
MB)
                           1.6/1.6 MB
11.7 MB/s eta 0:00:00
Downloading
MarkupSafe-2.1.3-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (25
kB)
Downloading
matplotlib-3.7.2-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl
(11.6 MB)
                           11.6/11.6 MB
76.0 MB/s eta 0:00:00
Downloading matplotlib_inline-0.1.6-py3-none-any.whl (9.4 kB)
```

```
Downloading mistune-3.0.1-py3-none-any.whl (47 kB)
                          48.0/48.0 kB
2.8 MB/s eta 0:00:00
Downloading nbclient-0.8.0-py3-none-any.whl (73 kB)
                          73.1/73.1 kB
5.4 MB/s eta 0:00:00
Downloading nbconvert-7.7.4-py3-none-any.whl (254 kB)
                          254.6/254.6 kB
18.4 MB/s eta 0:00:00
Downloading nbformat-5.9.2-py3-none-any.whl (77 kB)
                          77.6/77.6 kB
5.7 MB/s eta 0:00:00
Downloading nest_asyncio-1.5.7-py3-none-any.whl (5.3 kB)
Downloading notebook_shim-0.2.3-py3-none-any.whl (13 kB)
Downloading overrides-7.4.0-py3-none-any.whl (17 kB)
Downloading packaging-23.1-py3-none-any.whl (48 kB)
                          48.9/48.9 kB
3.1 MB/s eta 0:00:00
Downloading pandocfilters-1.5.0-py2.py3-none-any.whl (8.7 kB)
Downloading parso-0.8.3-py2.py3-none-any.whl (100 kB)
                          100.8/100.8 kB
7.0 MB/s eta 0:00:00
Downloading pexpect-4.8.0-py2.py3-none-any.whl (59 kB)
                          59.0/59.0 kB
4.2 MB/s eta 0:00:00
Downloading Pillow-10.0.0-cp310-cp310-manylinux_2_28_x86_64.whl (3.4 MB)
                          3.4/3.4 MB
61.8 MB/s eta 0:00:00
Downloading platformdirs-3.10.0-py3-none-any.whl (17 kB)
Downloading prometheus_client-0.17.1-py3-none-any.whl (60 kB)
                          60.6/60.6 kB
4.4 MB/s eta 0:00:00
Downloading prompt_toolkit-3.0.39-py3-none-any.whl (385 kB)
                          385.2/385.2 kB
24.8 MB/s eta 0:00:00
Downloading pure_eval-0.2.2-py3-none-any.whl (11 kB)
Downloading pycparser-2.21-py2.py3-none-any.whl (118 kB)
                          118.7/118.7 kB
8.6 MB/s eta 0:00:00
Downloading Pygments-2.16.1-py3-none-any.whl (1.2 MB)
                          1.2/1.2 MB
49.4 MB/s eta 0:00:00
Downloading pyparsing-3.0.9-py3-none-any.whl (98 kB)
                          98.3/98.3 kB
7.4 MB/s eta 0:00:00
Downloading python_json_logger-2.0.7-py3-none-any.whl (8.1 kB)
Downloading pytz-2023.3-py2.py3-none-any.whl (502 kB)
                          502.3/502.3 kB
```

```
Downloading
numpy-1.23.5-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (17.1
MB)
                              17.1/17.1 MB
83.1 MB/s eta 0:00:00
Downloading
pandas-1.5.3-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (12.1
MB)
                              12.1/12.1 MB
89.8 MB/s eta 0:00:00
Downloading tornado-6.3.2-cp38-abi3-manylinux_2_5_x86_64.manylinux1_x86_64
.manylinux_2_17_x86_64.manylinux2014_x86_64.whl (426 kB)
                              426.9/426.9 kB
23.9 MB/s eta 0:00:00
Installing collected packages: wcwidth, pytz, pure-eval, json5,
fastjsonschema, executing, appnope, websocket-client, webcolors, urllib3, uri-
template, tzdata, traitlets, tornado, tinycss2, soupsieve, sniffio, Send2Trash,
rpds-py, rfc3986-validator, rfc3339-validator, PyYAML, python-json-logger,
pyparsing, Pygments, pycparser, prompt-toolkit, prometheus-client, platformdirs,
Pillow, pexpect, parso, pandocfilters, packaging, overrides, numpy, nest-
asyncio, mistune, MarkupSafe, kiwisolver, jupyterlab-pygments, jsonpointer,
idna, fqdn, fonttools, debugpy, cycler, charset-normalizer, certifi, bleach,
Babel, attrs, async-lru, asttokens, terminado, stack-data, scipy, requests,
referencing, pandas, matplotlib-inline, jupyter_core, Jinja2, jedi, imageio,
contourpy, comm, cffi, beautifulsoup4, arrow, matplotlib,
jupyter_server_terminals, jupyter_client, jsonschema-specifications,
isoduration, seaborn, jsonschema, nbformat, nbclient, jupyter-events, nbconvert,
jupyter_server, notebook_shim, jupyterlab_server, jupyter-lsp, jupyterlab
  Attempting uninstall: wcwidth
    Found existing installation: wcwidth 0.2.13
    Uninstalling wcwidth-0.2.13:
      Successfully uninstalled wcwidth-0.2.13
  Attempting uninstall: pytz
    Found existing installation: pytz 2024.2
    Uninstalling pytz-2024.2:
      Successfully uninstalled pytz-2024.2
  Attempting uninstall: fastjsonschema
    Found existing installation: fastjsonschema 2.21.1
    Uninstalling fastjsonschema-2.21.1:
      Successfully uninstalled fastjsonschema-2.21.1
  Attempting uninstall: websocket-client
    Found existing installation: websocket-client 1.8.0
    Uninstalling websocket-client-1.8.0:
      Successfully uninstalled websocket-client-1.8.0
  Attempting uninstall: webcolors
    Found existing installation: webcolors 24.11.1
    Uninstalling webcolors-24.11.1:
      Successfully uninstalled webcolors-24.11.1
```

```
Attempting uninstall: urllib3
  Found existing installation: urllib3 2.2.3
  Uninstalling urllib3-2.2.3:
    Successfully uninstalled urllib3-2.2.3
Attempting uninstall: tzdata
  Found existing installation: tzdata 2024.2
  Uninstalling tzdata-2024.2:
    Successfully uninstalled tzdata-2024.2
Attempting uninstall: traitlets
  Found existing installation: traitlets 5.7.1
  Uninstalling traitlets-5.7.1:
    Successfully uninstalled traitlets-5.7.1
Attempting uninstall: tornado
  Found existing installation: tornado 6.3.3
  Uninstalling tornado-6.3.3:
    Successfully uninstalled tornado-6.3.3
Attempting uninstall: tinycss2
  Found existing installation: tinycss2 1.4.0
  Uninstalling tinycss2-1.4.0:
    Successfully uninstalled tinycss2-1.4.0
Attempting uninstall: soupsieve
  Found existing installation: soupsieve 2.6
  Uninstalling soupsieve-2.6:
    Successfully uninstalled soupsieve-2.6
Attempting uninstall: sniffio
  Found existing installation: sniffio 1.3.1
  Uninstalling sniffio-1.3.1:
    Successfully uninstalled sniffio-1.3.1
Attempting uninstall: Send2Trash
  Found existing installation: Send2Trash 1.8.3
  Uninstalling Send2Trash-1.8.3:
    Successfully uninstalled Send2Trash-1.8.3
Attempting uninstall: rpds-py
  Found existing installation: rpds-py 0.22.0
  Uninstalling rpds-py-0.22.0:
    Successfully uninstalled rpds-py-0.22.0
Attempting uninstall: PyYAML
  Found existing installation: PyYAML 6.0.2
  Uninstalling PyYAML-6.0.2:
    Successfully uninstalled PyYAML-6.0.2
Attempting uninstall: pyparsing
  Found existing installation: pyparsing 3.2.0
  Uninstalling pyparsing-3.2.0:
    Successfully uninstalled pyparsing-3.2.0
Attempting uninstall: Pygments
  Found existing installation: Pygments 2.18.0
  Uninstalling Pygments-2.18.0:
    Successfully uninstalled Pygments-2.18.0
```

```
Attempting uninstall: pycparser
  Found existing installation: pycparser 2.22
  Uninstalling pycparser-2.22:
    Successfully uninstalled pycparser-2.22
Attempting uninstall: prompt-toolkit
  Found existing installation: prompt_toolkit 3.0.48
  Uninstalling prompt_toolkit-3.0.48:
    Successfully uninstalled prompt_toolkit-3.0.48
Attempting uninstall: prometheus-client
  Found existing installation: prometheus_client 0.21.0
  Uninstalling prometheus_client-0.21.0:
    Successfully uninstalled prometheus_client-0.21.0
Attempting uninstall: platformdirs
  Found existing installation: platformdirs 4.3.6
  Uninstalling platformdirs-4.3.6:
    Successfully uninstalled platformdirs-4.3.6
Attempting uninstall: Pillow
  Found existing installation: pillow 11.0.0
  Uninstalling pillow-11.0.0:
    Successfully uninstalled pillow-11.0.0
Attempting uninstall: pexpect
  Found existing installation: pexpect 4.9.0
  Uninstalling pexpect-4.9.0:
    Successfully uninstalled pexpect-4.9.0
Attempting uninstall: parso
  Found existing installation: parso 0.8.4
  Uninstalling parso-0.8.4:
    Successfully uninstalled parso-0.8.4
Attempting uninstall: pandocfilters
  Found existing installation: pandocfilters 1.5.1
  Uninstalling pandocfilters-1.5.1:
    Successfully uninstalled pandocfilters-1.5.1
Attempting uninstall: packaging
  Found existing installation: packaging 24.2
  Uninstalling packaging-24.2:
    Successfully uninstalled packaging-24.2
Attempting uninstall: numpy
  Found existing installation: numpy 1.26.4
  Uninstalling numpy-1.26.4:
    Successfully uninstalled numpy-1.26.4
Attempting uninstall: nest-asyncio
  Found existing installation: nest-asyncio 1.6.0
  Uninstalling nest-asyncio-1.6.0:
    Successfully uninstalled nest-asyncio-1.6.0
Attempting uninstall: mistune
  Found existing installation: mistune 3.0.2
  Uninstalling mistune-3.0.2:
    Successfully uninstalled mistune-3.0.2
```

```
Attempting uninstall: MarkupSafe
  Found existing installation: MarkupSafe 3.0.2
  Uninstalling MarkupSafe-3.0.2:
    Successfully uninstalled MarkupSafe-3.0.2
Attempting uninstall: kiwisolver
  Found existing installation: kiwisolver 1.4.7
  Uninstalling kiwisolver-1.4.7:
    Successfully uninstalled kiwisolver-1.4.7
Attempting uninstall: jupyterlab-pygments
  Found existing installation: jupyterlab_pygments 0.3.0
  Uninstalling jupyterlab_pygments-0.3.0:
    Successfully uninstalled jupyterlab_pygments-0.3.0
Attempting uninstall: jsonpointer
  Found existing installation: jsonpointer 3.0.0
  Uninstalling jsonpointer-3.0.0:
    Successfully uninstalled jsonpointer-3.0.0
Attempting uninstall: idna
  Found existing installation: idna 3.10
  Uninstalling idna-3.10:
    Successfully uninstalled idna-3.10
Attempting uninstall: fonttools
  Found existing installation: fonttools 4.55.0
  Uninstalling fonttools-4.55.0:
    Successfully uninstalled fonttools-4.55.0
Attempting uninstall: debugpy
  Found existing installation: debugpy 1.8.0
  Uninstalling debugpy-1.8.0:
    Successfully uninstalled debugpy-1.8.0
Attempting uninstall: cycler
  Found existing installation: cycler 0.12.1
  Uninstalling cycler-0.12.1:
    Successfully uninstalled cycler-0.12.1
Attempting uninstall: charset-normalizer
  Found existing installation: charset-normalizer 3.4.0
  Uninstalling charset-normalizer-3.4.0:
    Successfully uninstalled charset-normalizer-3.4.0
Attempting uninstall: certifi
  Found existing installation: certifi 2024.8.30
  Uninstalling certifi-2024.8.30:
    Successfully uninstalled certifi-2024.8.30
Attempting uninstall: bleach
  Found existing installation: bleach 6.2.0
  Uninstalling bleach-6.2.0:
    Successfully uninstalled bleach-6.2.0
Attempting uninstall: Babel
  Found existing installation: babel 2.16.0
  Uninstalling babel-2.16.0:
    Successfully uninstalled babel-2.16.0
```

```
Attempting uninstall: attrs
  Found existing installation: attrs 24.2.0
  Uninstalling attrs-24.2.0:
    Successfully uninstalled attrs-24.2.0
Attempting uninstall: terminado
  Found existing installation: terminado 0.18.1
  Uninstalling terminado-0.18.1:
    Successfully uninstalled terminado-0.18.1
Attempting uninstall: scipy
  Found existing installation: scipy 1.13.1
  Uninstalling scipy-1.13.1:
    Successfully uninstalled scipy-1.13.1
Attempting uninstall: requests
  Found existing installation: requests 2.32.3
  Uninstalling requests-2.32.3:
    Successfully uninstalled requests-2.32.3
Attempting uninstall: referencing
  Found existing installation: referencing 0.35.1
  Uninstalling referencing-0.35.1:
    Successfully uninstalled referencing-0.35.1
Attempting uninstall: pandas
  Found existing installation: pandas 2.2.2
  Uninstalling pandas-2.2.2:
    Successfully uninstalled pandas-2.2.2
Attempting uninstall: matplotlib-inline
  Found existing installation: matplotlib-inline 0.1.7
  Uninstalling matplotlib-inline-0.1.7:
    Successfully uninstalled matplotlib-inline-0.1.7
Attempting uninstall: jupyter_core
  Found existing installation: jupyter_core 5.7.2
  Uninstalling jupyter_core-5.7.2:
    Successfully uninstalled jupyter_core-5.7.2
Attempting uninstall: Jinja2
  Found existing installation: Jinja2 3.1.4
  Uninstalling Jinja2-3.1.4:
    Successfully uninstalled Jinja2-3.1.4
Attempting uninstall: imageio
  Found existing installation: imageio 2.36.1
  Uninstalling imageio-2.36.1:
    Successfully uninstalled imageio-2.36.1
Attempting uninstall: contourpy
  Found existing installation: contourpy 1.3.1
  Uninstalling contourpy-1.3.1:
    Successfully uninstalled contourpy-1.3.1
Attempting uninstall: cffi
  Found existing installation: cffi 1.17.1
  Uninstalling cffi-1.17.1:
    Successfully uninstalled cffi-1.17.1
```

```
Attempting uninstall: beautifulsoup4
  Found existing installation: beautifulsoup4 4.12.3
  Uninstalling beautifulsoup4-4.12.3:
    Successfully uninstalled beautifulsoup4-4.12.3
Attempting uninstall: matplotlib
  Found existing installation: matplotlib 3.8.0
  Uninstalling matplotlib-3.8.0:
    Successfully uninstalled matplotlib-3.8.0
Attempting uninstall: jupyter_client
  Found existing installation: jupyter-client 6.1.12
  Uninstalling jupyter-client-6.1.12:
    Successfully uninstalled jupyter-client-6.1.12
Attempting uninstall: jsonschema-specifications
  Found existing installation: jsonschema-specifications 2024.10.1
  Uninstalling jsonschema-specifications-2024.10.1:
    Successfully uninstalled jsonschema-specifications-2024.10.1
Attempting uninstall: seaborn
  Found existing installation: seaborn 0.13.2
  Uninstalling seaborn-0.13.2:
    Successfully uninstalled seaborn-0.13.2
Attempting uninstall: jsonschema
  Found existing installation: jsonschema 4.23.0
  Uninstalling jsonschema-4.23.0:
    Successfully uninstalled jsonschema-4.23.0
Attempting uninstall: nbformat
  Found existing installation: nbformat 5.10.4
  Uninstalling nbformat-5.10.4:
    Successfully uninstalled nbformat-5.10.4
Attempting uninstall: nbclient
  Found existing installation: nbclient 0.10.1
  Uninstalling nbclient-0.10.1:
    Successfully uninstalled nbclient-0.10.1
Attempting uninstall: nbconvert
  Found existing installation: nbconvert 7.16.4
  Uninstalling nbconvert-7.16.4:
    Successfully uninstalled nbconvert-7.16.4
Attempting uninstall: jupyter_server
  Found existing installation: jupyter-server 1.24.0
  Uninstalling jupyter-server-1.24.0:
    Successfully uninstalled jupyter-server-1.24.0
Attempting uninstall: notebook_shim
  Found existing installation: notebook_shim 0.2.4
  Uninstalling notebook_shim-0.2.4:
    Successfully uninstalled notebook_shim-0.2.4
```

```
ERROR: pip's dependency resolver does not currently take into account all
the packages that are installed. This behaviour is the source of the following
dependency conflicts.
albucore 0.0.19 requires numpy>=1.24.4, but you have numpy 1.23.5 which is
incompatible.
albumentations 1.4.20 requires numpy>=1.24.4, but you have numpy 1.23.5 which is
incompatible.
bigframes 1.27.0 requires numpy>=1.24.0, but you have numpy 1.23.5 which is
incompatible.
bokeh 3.6.1 requires contourpy>=1.2, but you have contourpy 1.1.0 which is
incompatible.
chex 0.1.87 requires numpy>=1.24.1, but you have numpy 1.23.5 which is
incompatible.
cudf-cu12 24.10.1 requires pandas<2.2.3dev0,>=2.0, but you have pandas 1.5.3
which is incompatible.
google-colab 1.0.0 requires pandas==2.2.2, but you have pandas 1.5.3 which is
incompatible.
google-colab 1.0.0 requires requests==2.32.3, but you have requests 2.31.0 which
is incompatible.
google-colab 1.0.0 requires tornado==6.3.3, but you have tornado 6.3.2 which is
incompatible.
jax 0.4.33 requires numpy>=1.24, but you have numpy 1.23.5 which is
incompatible.
jaxlib 0.4.33 requires numpy>=1.24, but you have numpy 1.23.5 which is
incompatible.
langchain-core 0.3.21 requires packaging<25,>=23.2, but you have packaging 23.1
which is incompatible.
mizani 0.13.0 requires pandas>=2.2.0, but you have pandas 1.5.3 which is
incompatible.
plotnine 0.14.3 requires matplotlib>=3.8.0, but you have matplotlib 3.7.2 which
is incompatible.
plotnine 0.14.3 requires pandas>=2.2.0, but you have pandas 1.5.3 which is
incompatible.
pygit2 1.16.0 requires cffi>=1.17.0, but you have cffi 1.15.1 which is
incompatible.
scikit-image 0.24.0 requires imageio>=2.33, but you have imageio 2.31.1 which is
```

```
Pillow-10.0.0 PyYAML-6.0.1 Pygments-2.16.1 Send2Trash-1.8.2 appnope-0.1.3
arrow-1.2.3 asttokens-2.2.1 async-lru-2.0.4 attrs-23.1.0 beautifulsoup4-4.12.2
bleach-6.0.0 certifi-2023.7.22 cffi-1.15.1 charset-normalizer-3.2.0 comm-0.1.4
contourpy-1.1.0 cycler-0.11.0 debugpy-1.6.7.post1 executing-1.2.0
fastjsonschema-2.18.0 fonttools-4.42.1 fqdn-1.5.1 idna-3.4 imageio-2.31.1
isoduration-20.11.0 jedi-0.19.0 json5-0.9.14 jsonpointer-2.4 jsonschema-4.19.0
jsonschema-specifications-2023.7.1 jupyter-events-0.7.0 jupyter-lsp-2.2.0
jupyter_client-7.4.9 jupyter_core-5.3.1 jupyter_server-2.7.2
jupyter_server_terminals-0.4.4 jupyterlab-4.0.5 jupyterlab-pygments-0.2.2
jupyterlab_server-2.24.0 kiwisolver-1.4.5 matplotlib-3.7.2 matplotlib-
inline-0.1.6 mistune-3.0.1 nbclient-0.8.0 nbconvert-7.7.4 nbformat-5.9.2 nest-
asyncio-1.5.7 notebook_shim-0.2.3 numpy-1.23.5 overrides-7.4.0 packaging-23.1
pandas-1.5.3 pandocfilters-1.5.0 parso-0.8.3 pexpect-4.8.0 platformdirs-3.10.0
prometheus-client-0.17.1 prompt-toolkit-3.0.39 pure-eval-0.2.2 pycparser-2.21
pyparsing-3.0.9 python-json-logger-2.0.7 pytz-2023.3 referencing-0.30.2
requests-2.31.0 rfc3339-validator-0.1.4 rfc3986-validator-0.1.1 rpds-py-0.9.2
scipy-1.11.2 seaborn-0.12.2 sniffio-1.3.0 soupsieve-2.4.1 stack-data-0.6.2
terminado-0.17.1 tinycss2-1.2.1 tornado-6.3.2 traitlets-5.9.0 tzdata-2023.3 uri-
template-1.3.0 urllib3-2.0.4 wcwidth-0.2.6 webcolors-1.13 websocket-client-1.6.2
```

1 Multiclass Support Vector Machine exercise

Complete and hand in this completed worksheet (including its outputs and any supporting code outside of the worksheet) with your assignment submission. For more details see the assignments page on the course website.

In this exercise you will:

- implement a fully-vectorized **loss function** for the SVM
- implement the fully-vectorized expression for its **analytic gradient**
- **check your implementation** using numerical gradient
- use a validation set to **tune the learning rate and regularization** strength
- **optimize** the loss function with **SGD**
- **visualize** the final learned weights

```python
# Run some setup code for this notebook.
from __future__ import print_function
import random
import numpy as np
from cs6353.data_utils import load_CIFAR10
import matplotlib.pyplot as plt



# This is a bit of magic to make matplotlib figures appear inline in the
# notebook rather than in a new window.
%matplotlib inline
plt.rcParams['figure.figsize'] = (10.0, 8.0) # set default size of plots
plt.rcParams['image.interpolation'] = 'nearest'
```

```
plt.rcParams['image.cmap'] = 'gray'


# Some more magic so that the notebook will reload external python modules;
# see http://stackoverflow.com/questions/1907993/
  ↪autoreload-of-modules-in-ipython
%load_ext autoreload
%autoreload 2
```

1.1 CIFAR-10 Data Loading and Preprocessing

```
[ ]: # Load the raw CIFAR-10 data.
     cifar10_dir = 'cs6353/datasets/cifar-10-batches-py'

     # Cleaning up variables to prevent loading data multiple times (which may cause␣
       ↪memory issue)
     try:
        del X_train, y_train
        del X_test, y_test
        print('Clear previously loaded data.')
     except:
        pass

     X_train, y_train, X_test, y_test = load_CIFAR10(cifar10_dir)

     # As a sanity check, we print out the size of the training and test data.
     print('Training data shape: ', X_train.shape)
     print('Training labels shape: ', y_train.shape)
     print('Test data shape: ', X_test.shape)
     print('Test labels shape: ', y_test.shape)
```

```
Training data shape:  (50000, 32, 32, 3)
Training labels shape:  (50000,)
Test data shape:  (10000, 32, 32, 3)
Test labels shape:  (10000,)
```

```
[ ]: # Visualize some examples from the dataset.
     # We show a few examples of training images from each class.
     classes = ['plane', 'car', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse',␣
       ↪'ship', 'truck']
     num_classes = len(classes)
     samples_per_class = 7
     for y, cls in enumerate(classes):
        idxs = np.flatnonzero(y_train == y)
        idxs = np.random.choice(idxs, samples_per_class, replace=False)
        for i, idx in enumerate(idxs):
            plt_idx = i * num_classes + y + 1
            plt.subplot(samples_per_class, num_classes, plt_idx)
```

```
        plt.imshow(X_train[idx].astype('uint8'))
        plt.axis('off')
        if i == 0:
            plt.title(cls)
plt.show()
```


plane   car   bird   cat   deer   dog   frog   horse   ship   truck
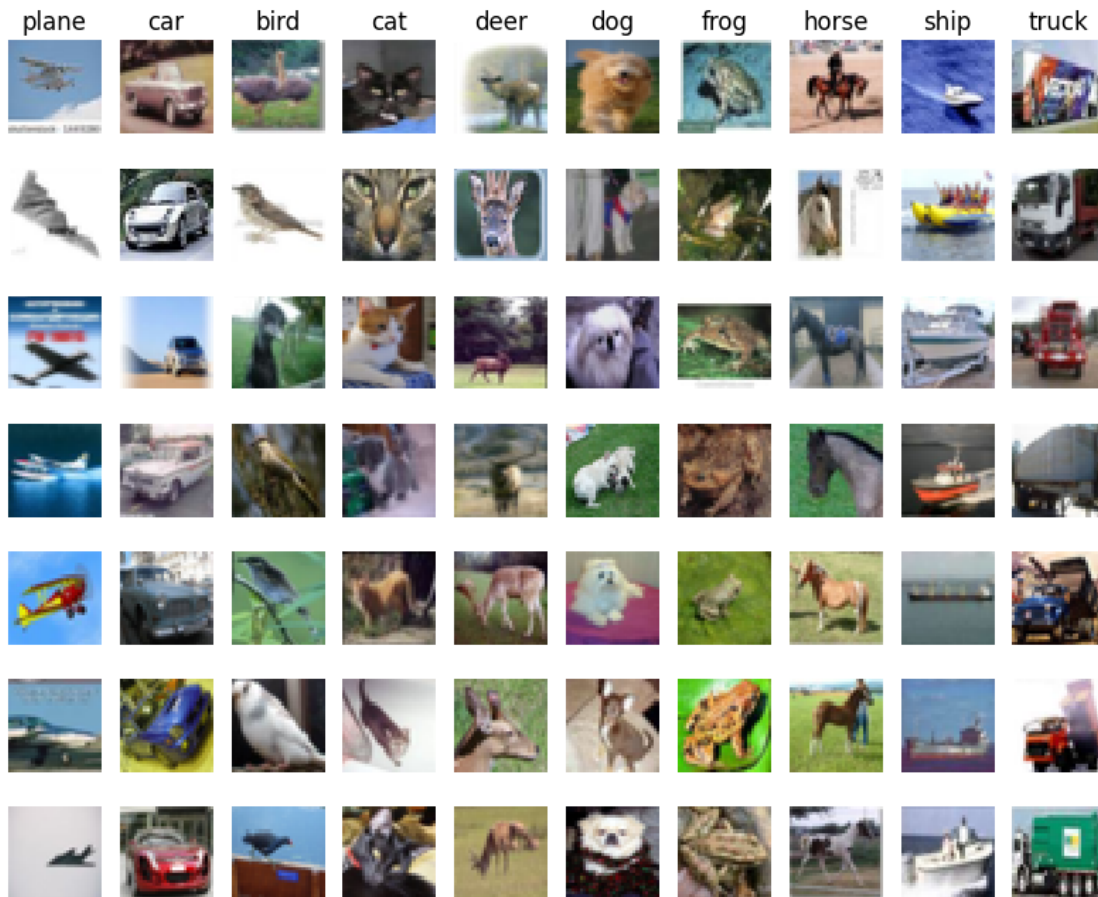
```
[ ]: # Split the data into train, val, and test sets. In addition we will
     # create a small development set as a subset of the training data;
     # we can use this for development so our code runs faster.
     num_training = 49000
     num_validation = 1000
     num_test = 1000
     num_dev = 500

     # Ensure we do not exceed the available size of X_train
     if X_train.shape[0] < num_training + num_validation:
         num_training = X_train.shape[0] - num_validation
```

```python
# Our validation set will be num_validation points from the original
# training set.
mask = range(num_training, num_training + num_validation)
X_val = X_train[list(mask)]  # Convert range to list to avoid slicing issues
y_val = y_train[list(mask)]

# Our training set will be the first num_train points from the original
# training set.
mask = range(num_training)
X_train = X_train[list(mask)]  # Convert range to list to avoid slicing issues
y_train = y_train[list(mask)]

# We will also make a development set, which is a small subset of
# the training set.
mask = np.random.choice(num_training, num_dev, replace=False)
X_dev = X_train[mask]
y_dev = y_train[mask]

# We use the first num_test points of the original test set as our
# test set.
X_test = X_test[:num_test]
y_test = y_test[:num_test]

print('Train data shape: ', X_train.shape)
print('Train labels shape: ', y_train.shape)
print('Validation data shape: ', X_val.shape)
print('Validation labels shape: ', y_val.shape)
print('Test data shape: ', X_test.shape)
print('Test labels shape: ', y_test.shape)
```

```
Train data shape:  (48000, 3074)
Train labels shape:  (48000,)
Validation data shape:  (1000, 3074)
Validation labels shape:  (1000,)
Test data shape:  (1000, 3075)
Test labels shape:  (1000,)
```

```python
# Preprocessing: reshape the image data into rows
X_train = np.reshape(X_train, (X_train.shape[0], -1))
X_val = np.reshape(X_val, (X_val.shape[0], -1))
X_test = np.reshape(X_test, (X_test.shape[0], -1))
X_dev = np.reshape(X_dev, (X_dev.shape[0], -1))

# Function to add the bias term only if it's not already present
def add_bias_term(X):
    if X.shape[1] == 3074:  # Check if bias term is missing
```

```
        return np.hstack([X, np.ones((X.shape[0], 1))])
    return X  # Return as-is if bias term is already added

# Add the bias term to all datasets
X_train = add_bias_term(X_train)
X_val = add_bias_term(X_val)
X_test = add_bias_term(X_test)
X_dev = add_bias_term(X_dev)

# As a sanity check, print out the shapes of the data
print('Training data shape: ', X_train.shape)
print('Validation data shape: ', X_val.shape)
print('Test data shape: ', X_test.shape)
print('dev data shape: ', X_dev.shape)
```

```
Training data shape:  (48000, 3076)
Validation data shape:  (1000, 3076)
Test data shape:  (1000, 3076)
dev data shape:  (500, 3076)
```

```
[ ]: # Preprocessing: subtract the mean image
     # first: compute the image mean based on the training data

     mean_image = np.mean(X_train[:, :-1], axis=0)  # Exclude the bias term
     print(mean_image[:10])  # Print a few of the elements

     # Rescale the mean image to [0, 255] for visualization
     mean_image_rescaled = 255.0 * (mean_image - np.min(mean_image)) / (np.
      ↪max(mean_image) - np.min(mean_image))

     # Visualize the mean image
     plt.figure(figsize=(4, 4))
     plt.imshow(mean_image_rescaled[:3072].reshape((32, 32, 3)).astype('uint8'))  #␣
      ↪Use only pixel values
     plt.title("Mean Image")
     plt.show()
```
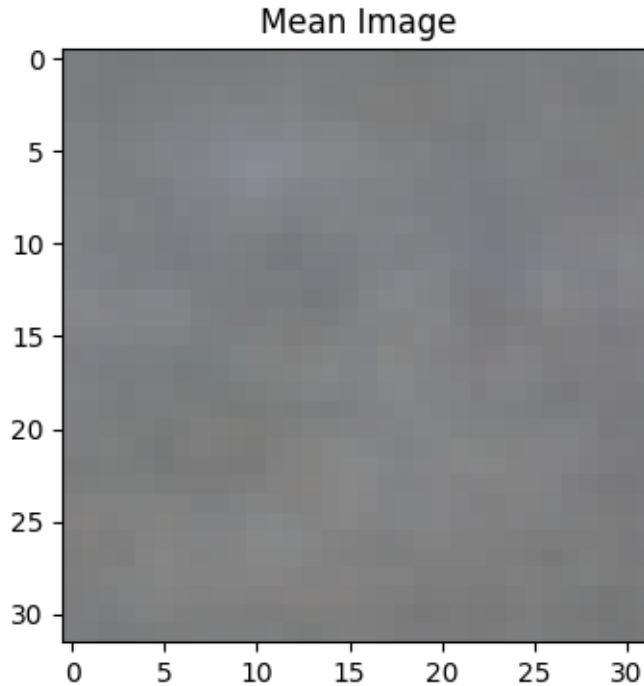
```
[-0.04239371 -0.02657781 -0.01770663 -0.04159779 -0.02608418 -0.01818793
 -0.06553231 -0.04908929 -0.03342432 -0.0712432 ]
```

Mean Image

```
[ ]:  # second: subtract the mean image from train and test data
      X_train[:, :-1] -= mean_image  # Exclude the bias term while subtracting the
       ↪mean
      X_val[:, :-1] -= mean_image
      X_test[:, :-1] -= mean_image
      X_dev[:, :-1] -= mean_image
```

```
[ ]:  # third: append the bias dimension of ones (i.e. bias trick) so that our SVM
      # only has to worry about optimizing a single weight matrix W.
      X_train = np.hstack([X_train, np.ones((X_train.shape[0], 1))])
      X_val = np.hstack([X_val, np.ones((X_val.shape[0], 1))])
      X_test = np.hstack([X_test, np.ones((X_test.shape[0], 1))])
      X_dev = np.hstack([X_dev, np.ones((X_dev.shape[0], 1))])

      print(X_train.shape, X_val.shape, X_test.shape, X_dev.shape)
```

(48000, 3077) (1000, 3077) (1000, 3077) (500, 3077)

1.2 SVM Classifier

Your code for this section will all be written inside **cs6353/classifiers/linear_svm.py**.

As you can see, we have prefilled the function `svm_loss_naive` which uses for loops to evaluate the multiclass SVM loss function.

```python
# Evaluate the naive implementation of the loss we provided for you:
from cs6353.classifiers.linear_svm import svm_loss_naive
import time

# Fix the bias term for all datasets to ensure it's added only once
def ensure_bias_term(X):
    if X.shape[1] == 3074:  # Add bias term if missing
        return np.hstack([X, np.ones((X.shape[0], 1))])
    elif X.shape[1] > 3075:  # Fix if multiple bias terms were added
        return np.hstack([X[:, :3074], np.ones((X.shape[0], 1))])
    return X  # Return as-is if already correct

# Apply the fix
X_train = ensure_bias_term(X_train)
X_val = ensure_bias_term(X_val)
X_test = ensure_bias_term(X_test)
X_dev = ensure_bias_term(X_dev)




# generate a random SVM weight matrix of small numbers
W = np.random.randn(X_dev.shape[1], 10) * 0.0001 # Match the number of features
 ↪in X_dev

loss, grad = svm_loss_naive(W, X_dev, y_dev, 0.000005)
print('loss: %f' % (loss, ))
```

loss: 9.131110

The `grad` returned from the function above is right now all zero. Derive and implement the gradient for the SVM cost function and implement it inline inside the function `svm_loss_naive`. You will find it helpful to interleave your new code inside the existing function.

To check that you have correctly implemented the gradient correctly, you can numerically estimate the gradient of the loss function and compare the numeric estimate to the gradient that you computed. We have provided code that does this for you:

```python
# Once you've implemented the gradient, recompute it with the code below
# and gradient check it with the function we provided for you

# Compute the loss and its gradient at W.
loss, grad = svm_loss_naive(W, X_dev, y_dev, 0.0)

# Numerically compute the gradient along several randomly chosen dimensions, and
# compare them with your analytically computed gradient. The numbers should
 ↪match
# almost exactly along all dimensions.
```

```python
from cs6353.gradient_check import grad_check_sparse
f = lambda w: svm_loss_naive(w, X_dev, y_dev, 0.0)[0]
grad_numerical = grad_check_sparse(f, W, grad)

# do the gradient check once again with regularization turned on
# you didn't forget the regularization gradient did you?
loss, grad = svm_loss_naive(W, X_dev, y_dev, 5e1)
f = lambda w: svm_loss_naive(w, X_dev, y_dev, 5e1)[0]
grad_numerical = grad_check_sparse(f, W, grad)
```

```
numerical: 7.627152 analytic: 7.627152, relative error: 7.930395e-11
numerical: -4.559018 analytic: -4.559018, relative error: 9.141636e-12
numerical: 23.104633 analytic: 23.104633, relative error: 1.024560e-11
numerical: 6.524187 analytic: 6.524187, relative error: 1.198299e-11
numerical: 44.017363 analytic: 43.940278, relative error: 8.763782e-04
numerical: 0.460760 analytic: 0.460760, relative error: 2.001671e-09
numerical: 7.207787 analytic: 7.207787, relative error: 4.673573e-11
numerical: -0.406687 analytic: -0.406687, relative error: 4.213492e-10
numerical: -5.603932 analytic: -5.603932, relative error: 9.727154e-11
numerical: 25.428283 analytic: 25.378575, relative error: 9.783778e-04
numerical: -12.946339 analytic: -12.946339, relative error: 4.031426e-11
numerical: 17.912832 analytic: 17.912832, relative error: 5.833780e-12
numerical: 34.680593 analytic: 34.680593, relative error: 4.300267e-12
numerical: 5.969931 analytic: 5.969931, relative error: 2.150919e-11
numerical: -4.295883 analytic: -4.295883, relative error: 1.274315e-10
numerical: 12.341549 analytic: 12.341549, relative error: 1.361787e-11
numerical: 5.359410 analytic: 5.411949, relative error: 4.877659e-03
numerical: 3.840894 analytic: 3.840894, relative error: 1.012976e-11
numerical: -6.250276 analytic: -6.263220, relative error: 1.034421e-03
numerical: 14.978978 analytic: 14.978978, relative error: 1.176466e-11
```

1.2.1 Inline Question 1:

It is possible that once in a while a dimension in the gradient check will not match exactly. What could such a discrepancy be caused by? Is it a reason for concern? What is a simple example in one dimension where a gradient check could fail? How would change the margin affect of the frequency of this happening? *Hint: the SVM loss function is not strictly speaking differentiable*

Your Answer: *fill this in.

PART1: Why might a dimension in the gradient check not match exactly?

Sometimes, when you compare the numerically computed gradient to the analytically computed gradient, the two won't match perfectly. This can happen for two main reasons:

1. Numerical Precision Issues: Since Gradient checking uses a numerical approximation based on small changes to the weights, like this:

   f/ W   (f(W+h)-f(W-h))/2h.

Thus if h (the step size) is too small, rounding errors in the floating-point arithmetic can mess up

the result. These errors don't usually mean your gradient is wrong but are just a limitation of the numerical method

    2. Non-Differentiability: The SVM loss function has "kinks" at certain points where it switches between being flat (loss is zero) and having a positive slope (loss is greater than zero). These kinks occur when the margin is exactly zero, as in:

margin = max(0, s - s +)

At these points, the function is not smooth, and its gradient is undefined. Numerical methods might give you a value, but there's no single "correct" answer here because the gradient doesn't technically exist.

Part 2: Should this be a reason for concern?

I don't believe so small mismatches are common and expected due to numerical precision and the nature of the loss function. They're only a problem if they happen in many places or the errors are large. If most dimensions match well and the relative errors are small, your gradient implementation is likely fine.

Part 3: Can you give a simple example where gradient checks fail?

Let's look at the one-dimensional function:

```
f(x) = max(0,x)
```

```
* if x > 0, gradient is f'(x) = 1.
* if x < 0, gradient f'(x) = 0.
* However at x = 0, gradient would be undefined due to the function suddenly switches from fla
```

Thus, if we use gradient at x = 0. We would have:

```
f'(x)   (f(h) - f(-h))/2h = (h-0)/2h   0.5
```

As a reuslt the numerical gradient 0.5 would not match with the analytical gradient (undefined), which shows why the check might fail.

Part 4: How does the margin size affect this?

- Larger Margin (> 1): The function becomes smoother because the margin is less likely to reach zero. Discrepancies in gradient checks become less frequent.

- Smaller Margin (< 1): The function has more kinks because the margin reaches zero more often. This makes discrepancies more frequent.

```
[ ]:  # Next implement the function svm_loss_vectorized; for now only compute the␣
      ↪loss;
      # we will implement the gradient in a moment.
      tic = time.time()
      loss_naive, grad_naive = svm_loss_naive(W, X_dev, y_dev, 0.000005)
      toc = time.time()
      print('Naive loss: %e computed in %fs' % (loss_naive, toc - tic))

      from cs6353.classifiers.linear_svm import svm_loss_vectorized
```

```
tic = time.time()
loss_vectorized, _ = svm_loss_vectorized(W, X_dev, y_dev, 0.000005)
toc = time.time()
print('Vectorized loss: %e computed in %fs' % (loss_vectorized, toc - tic))


# The losses should match but your vectorized implementation should be much␣
 ↪faster.
print('difference: %f' % (loss_naive - loss_vectorized))
```

```
Naive loss: 9.131110e+00 computed in 0.226418s
Vectorized loss: 9.131110e+00 computed in 0.026713s
difference: -0.000000
```

```
[ ]: # Complete the implementation of svm_loss_vectorized, and compute the gradient
     # of the loss function in a vectorized way.

     # The naive implementation and the vectorized implementation should match, but
     # the vectorized version should still be much faster.
     tic = time.time()
     _, grad_naive = svm_loss_naive(W, X_dev, y_dev, 0.000005)
     toc = time.time()
     print('Naive loss and gradient: computed in %fs' % (toc - tic))

     tic = time.time()
     _, grad_vectorized = svm_loss_vectorized(W, X_dev, y_dev, 0.000005)
     toc = time.time()
     print('Vectorized loss and gradient: computed in %fs' % (toc - tic))

     # The loss is a single number, so it is easy to compare the values computed
     # by the two implementations. The gradient on the other hand is a matrix, so
     # we use the Frobenius norm to compare them.
     difference = np.linalg.norm(grad_naive - grad_vectorized, ord='fro')
     print('difference: %f' % difference)
```

```
Naive loss and gradient: computed in 0.141822s
Vectorized loss and gradient: computed in 0.010362s
difference: 0.000000
```

1.2.2   Stochastic Gradient Descent

We now have vectorized and efficient expressions for the loss, the gradient and our gradient matches the numerical gradient. We are therefore ready to do SGD to minimize the loss. Your code for this part will be written inside cs6353/classifiers/linear_classifier.py.

```
[ ]: # In the file linear_classifier.py, implement SGD in the function
     # LinearClassifier.train() and then run it with the code below.
     from cs6353.classifiers import LinearSVM
     svm = LinearSVM()
```

```
tic = time.time()
loss_hist = svm.train(X_train, y_train, learning_rate=1e-7, reg=2.5e4,
                      num_iters=1500, verbose=True)
toc = time.time()
print('That took %fs' % (toc - tic))
```

```
iteration 0 / 1500: loss 792.811469
iteration 100 / 1500: loss 288.278937
iteration 200 / 1500: loss 107.876980
iteration 300 / 1500: loss 43.021308
iteration 400 / 1500: loss 18.195767
iteration 500 / 1500: loss 9.912935
iteration 600 / 1500: loss 7.101565
iteration 700 / 1500: loss 5.658936
iteration 800 / 1500: loss 5.321759
iteration 900 / 1500: loss 5.943075
iteration 1000 / 1500: loss 5.132132
iteration 1100 / 1500: loss 4.993031
iteration 1200 / 1500: loss 5.354704
iteration 1300 / 1500: loss 5.622773
iteration 1400 / 1500: loss 5.562087
That took 11.217257s
```

```
[ ]: # A useful debugging strategy is to plot the loss as a function of
     # iteration number:
     plt.plot(loss_hist)
     plt.xlabel('Iteration number')
     plt.ylabel('Loss value')
     plt.show()
```
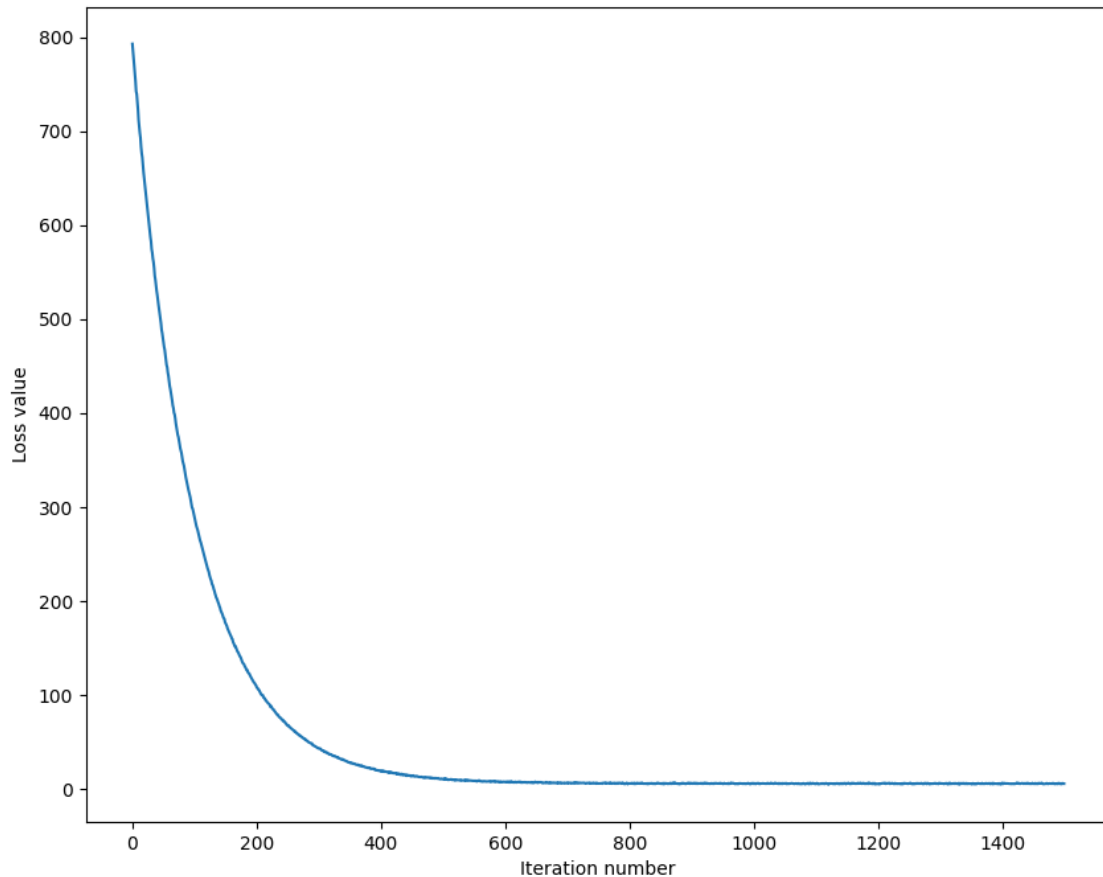
```
[ ]: # Write the LinearSVM.predict function and evaluate the performance on both the
     # training and validation set
     y_train_pred = svm.predict(X_train)
     print('training accuracy: %f' % (np.mean(y_train == y_train_pred), ))
     y_val_pred = svm.predict(X_val)
     print('validation accuracy: %f' % (np.mean(y_val == y_val_pred), ))
```

```
training accuracy: 0.367792
validation accuracy: 0.335000
```

```
[115]: # Use the validation set to tune hyperparameters (regularization strength and
       # learning rate). You should experiment with different ranges for the learning
       # rates and regularization strengths; if you are careful you should be able to
       # get a classification accuracy of about 0.4 on the validation set.
       learning_rates = [0.5e-7, 1e-7, 2e-7]
       regularization_strengths = [2.0e4, 2.5e4, 3.0e4]

       # results is dictionary mapping tuples of the form
       # (learning_rate, regularization_strength) to tuples of the form
```

```python
# (training_accuracy, validation_accuracy). The accuracy is simply the fraction
# of data points that are correctly classified.
results = {}
best_val = -1   # The highest validation accuracy that we have seen so far.
best_svm = None # The LinearSVM object that achieved the highest validation
  rate.


################################################################################
# TODO:                                                                        #
# Write code that chooses the best hyperparameters by tuning on the validation #
# set. For each combination of hyperparameters, train a linear SVM on the      #
# training set, compute its accuracy on the training and validation sets, and  #
# store these numbers in the results dictionary. In addition, store the best   #
# validation accuracy in best_val and the LinearSVM object that achieves this  #
# accuracy in best_svm.                                                        #
#                                                                              #
# Hint: You should use a small value for num_iters as you develop your         #
# validation code so that the SVMs don't take much time to train; once you are #
# confident that your validation code works, you should rerun the validation   #
# code with a larger value for num_iters.                                      #
################################################################################

# Iterate over all combinations of learning rates and regularization strengths
for lr in learning_rates:
    for reg in regularization_strengths:
        svm = LinearSVM()
        svm.train(X_train, y_train, learning_rate=lr, reg=reg, num_iters=1500,
  verbose=False)   # Use lr and reg

        y_train_pred = svm.predict(X_train)
        train_accuracy = np.mean(y_train_pred == y_train)

        y_val_pred = svm.predict(X_val)
        val_accuracy = np.mean(y_val_pred == y_val)

        results[(lr, reg)] = (train_accuracy, val_accuracy)
        if val_accuracy > best_val:
            best_val = val_accuracy
            best_svm = svm



################################################################################
#                            END OF YOUR CODE                                  #
################################################################################

# Print out results.
for lr, reg in results:
```

```
    train_acc, val_acc = results[(lr, reg)]
    print(f'lr {lr:e} reg {reg:e} train accuracy: {train_acc:.3f} val accuracy:␣
 ↪{val_acc:.3f}')

print(f'Best validation accuracy: {best_val:.3f}')
```

```
lr 5.000000e-08 reg 2.000000e+04 train accuracy: 0.375 val accuracy: 0.346
lr 5.000000e-08 reg 2.500000e+04 train accuracy: 0.373 val accuracy: 0.348
lr 5.000000e-08 reg 3.000000e+04 train accuracy: 0.366 val accuracy: 0.342
lr 1.000000e-07 reg 2.000000e+04 train accuracy: 0.373 val accuracy: 0.342
lr 1.000000e-07 reg 2.500000e+04 train accuracy: 0.372 val accuracy: 0.342
lr 1.000000e-07 reg 3.000000e+04 train accuracy: 0.366 val accuracy: 0.346
lr 2.000000e-07 reg 2.000000e+04 train accuracy: 0.357 val accuracy: 0.328
lr 2.000000e-07 reg 2.500000e+04 train accuracy: 0.365 val accuracy: 0.348
lr 2.000000e-07 reg 3.000000e+04 train accuracy: 0.360 val accuracy: 0.340
Best validation accuracy: 0.348
```

[106]:
```python
# Visualize the cross-validation results
import math
import matplotlib.pyplot as plt


x_scatter = [math.log10(x[0]) for x in results]  # log of learning rates
y_scatter = [math.log10(x[1]) for x in results]  # log of regularization␣
 ↪strengths

# plot training accuracy
marker_size = 100  # Adjust marker size as necessary
colors = [results[x][0] for x in results]  # Training accuracy
plt.figure(figsize=(10, 8))
plt.subplot(2, 1, 1)
plt.scatter(x_scatter, y_scatter, marker_size, c=colors, cmap='viridis')
plt.colorbar(label='Training Accuracy')
plt.xlabel('Log Learning Rate')
plt.ylabel('Log Regularization Strength')
plt.title('CIFAR-10 Training Accuracy')


# plot validation accuracy
colors = [results[x][1] for x in results]  # Validation accuracy
plt.subplot(2, 1, 2)
plt.scatter(x_scatter, y_scatter, marker_size, c=colors, cmap='viridis')
plt.colorbar(label='Validation Accuracy')
plt.xlabel('Log Learning Rate')
plt.ylabel('Log Regularization Strength')
plt.title('CIFAR-10 Validation Accuracy')
```
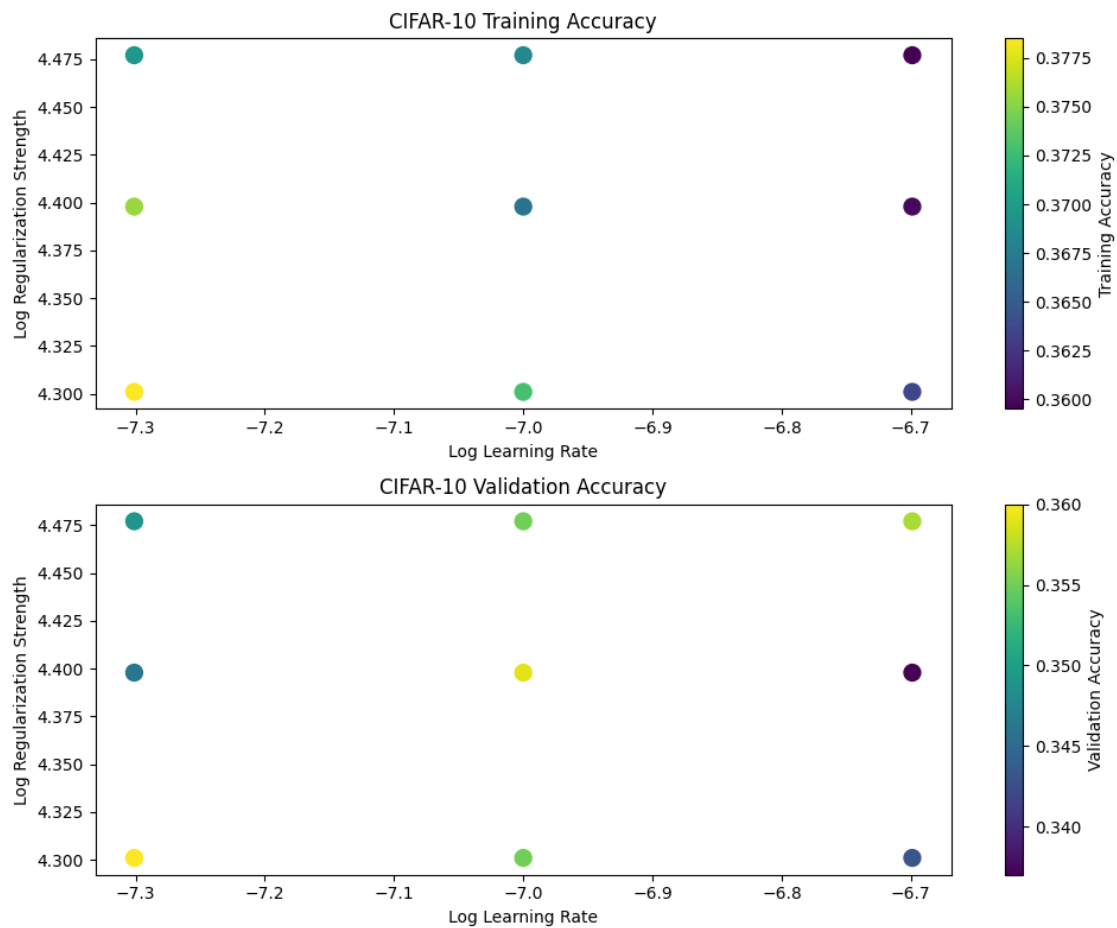
```
plt.tight_layout()
plt.show()
```

CIFAR-10 Training Accuracy

CIFAR-10 Validation Accuracy

[110]:
```
# Evaluate the best svm on test set
y_test_pred = best_svm.predict(X_test)
test_accuracy = np.mean(y_test == y_test_pred)
print('linear SVM on raw pixels final test set accuracy: %f' % test_accuracy)
```

linear SVM on raw pixels final test set accuracy: 0.386000

[114]:
```
# Visualize the learned weights for each class.
# Depending on your choice of learning rate and regularization strength, these␣
 ↪may
# or may not be nice to look at.

print("Shape of W:", best_svm.W.shape)
```

```python
# Strip out the bias and ensure proper dimensions
w = best_svm.W[:-1, :]  # Remove the bias term

# Check if the weights have the correct shape for visualization
if w.shape[0] == 32 * 32 * 3:
    w = w.reshape(32, 32, 3, 10)  # Reshape to (32, 32, 3, 10)
elif w.shape[0] > 32 * 32 * 3:
    # Truncate to match expected image size
    print(f"Truncating weights from shape {w.shape} to {(32 * 32 * 3, 10)}")
    w = w[:32 * 32 * 3, :].reshape(32, 32, 3, 10)
else:
    raise ValueError(f"Unexpected weight matrix shape: {w.shape}. Check input
 ↪dimensions.")

# Normalize the weights to [0, 255] for visualization
w_min, w_max = np.min(w), np.max(w)
classes = ['plane', 'car', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse',
 ↪'ship', 'truck']

plt.figure(figsize=(12, 6))
for i in range(10):
    plt.subplot(2, 5, i + 1)
    wimg = 255.0 * (w[:, :, :, i].squeeze() - w_min) / (w_max - w_min)  #
 ↪Rescale weights
    plt.imshow(wimg.astype('uint8'))
    plt.axis('off')
    plt.title(classes[i])
plt.show()
```
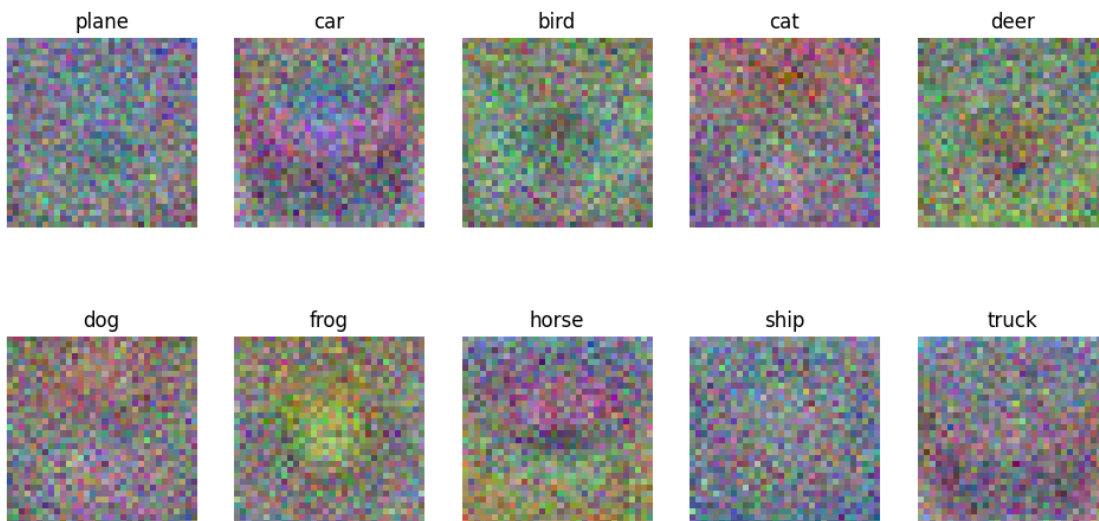
```
Shape of W: (3075, 10)
Truncating weights from shape (3074, 10) to (3072, 10)
```

1.2.3 Inline question 2:

Describe what your visualized SVM weights look like, and offer a brief explanation for why they look the way that they do.

Your answer: *fill this in*

1. Linear SVM Nature:

   - The SVM learns a single linear classifier for each class, which acts as a weighted sum of pixel values. This means the weights represent the importance of each pixel in predicting the presence of a specific class.

   - However, because the decision boundaries are linear, the SVM cannot capture complex non-linear features such as shapes, textures, or higher-order patterns that a deep neural network could.

2. Raw Pixel Input:

   - The weights are derived directly from raw pixel data. Since CIFAR-10 images are small (32x32) and contain complex scenes, the SVM struggles to identify meaningful features at this level. Instead, it assigns weights based on the average pixel differences between classes, leading to the noisy, abstract patterns.

3. Class Overlap:

   - Many CIFAR-10 classes share common background features or have significant visual overlap (e.g., "cat" and "dog" or "ship" and "plane"). The SVM attempts to differentiate these classes but may assign weights to irrelevant regions because it cannot effectively isolate shared or background features.

4. Regularization Effects:

   - The regularization term (L2) in the SVM loss function prevents the weights from becoming too large. This enforces smoothness but may also reduce the ability of the SVM to focus heavily on specific features.

5. Data Complexity vs. Model Simplicity:

- The CIFAR-10 dataset contains high variability and noise due to its diverse classes and relatively small resolution. A linear model like SVM, which assumes simple linear separability, is fundamentally limited in capturing these complex variations, resulting in weight maps that look noisy and abstract.

Thus, the SVM weights look like noisy, colorful gradients because the model is working directly with raw pixel data and trying to learn linear boundaries between classes. While these weights give some insight into what the SVM focuses on for each class, they're not interpretable as clear features due to the simplicity of the model and the complexity of the dataset. This highlights why more advanced techniques, like feature extraction or neural networks, are often needed to handle image data effectively.