با نام پروردگار

دانشگاه صنعتی امیرکبیر (پلی‌تکنیک تهران)

درس پردازش دادگان حجیم

# تحلیل انسانی، دسته‌بندی و خوشه‌بندی یادگیری‌ماشینی دادگان حجیم و انجام چند عملیات ریاضی به روش نگاشت‌کاهش


استاد : دکتر پورفرد

سرپرست تیم استادیاری: دکتر طالبی

استادیار طراح تمرین: نیما موسوی منفرد


تمرین شماره ۱

پاییز ۱۴۰۴

**دانشی که با انجام این سری تمرین می‌آموزیم:**

- وارسی دادگان و دسته بندی دادگان به کمک روشهای پردازش دادگان حجیم
- کار با دادگان بدون برچسب و خوشه‌بندی به کمک روش‌های پردازش دادگان حجیم
- شمارش واژگان و محاسبه میانگین دما به شیوه نگاشت-کاهش
- انجام عملیات ضرب و جمع ماتریسی به شیوه نگاشت-کاهش

**مهارت‌هایی که با انجام این سری تمرین می‌آموزیم:**

- تقویت برنامه‌نویسی با زبان Python
- کار با PySpark یا رابط پایتونی فریمورک Apache Spark
- کار با MRJob کتابخانه پایتونی پیاده‌سازی MapReduce
- پیاده‌سازی Classification (Dession tree) به کمک PySpark
- پیاده‌سازی Clustring (K-Means++)به کمک PySpark
- پیاده‌سازی الگوریتم‌های مبتنی بر MapReduce به کمک PySpark و MRJob

**نکات مفید برای شروع:**

- هدف از این تمرین‌ها ارتقا دانش و مهارت شماست.
- از دوستان خود و هوش مصنوعی راهنمایی و کمک بگیرید. اما مسئولیت کدها، نتایج و تحلیل‌ها برعهده شماست.
- پس دانش خود را ارتقا دهید و تلاش کنید دانسته خود را در کد و گزارش ارائه دهید.
- برخلاف ظاهر طولانی سوالات، حجم واقعی و زمان مورد نیاز این سوالات کوتاه است.
  حجم بیشتر هر سوال را توضیحات گام به گام و راهنمایی‌ها در بر گرفته است.
- خرد کردن مسئله و راهنمایی گام به گام برای تسهیل فرآیند پاسخ‌گویی و رفع ابهامات احتمالی نوشته شده است. لطفا به راهنمایی‌ها توجه نمایید.
- همان مقدار که به انجام کامل و صحیح سوالات احتمام دارید، از تردید دست کشیده و تلاش کنید، تنها موارد خواسته شده را تکمیل، اجرا و گزارش کنید و در صورت وجود زمان بیشتر در انتها کار به ارتقا کار بپردازید.
- در این سری تمرین، اجرا الگوریتمها و مدلها در قالب روشهای پردازش دادگان حجیم انجام می‌شود. اما پس از اتمام پروسه اصلی، برای بخشهای جانبی مانند رسم نمودارها، پرینت نتایج می‌توانید از کتابخانه های دیگری Numpy، Pandas ، Matplotlib و... استفاده نمایید.
- برای شروع ابتدا پیوست را مطالعه نمایید.

**کلمات کلیدی:**

Big data analytics, PySpark, MRJob, MapReduce,Classification, Clustering, K-Means++, AI-Generated Texts, Matrix Addition and Multiplication, Average Temperature

# سوال۱)

امروزه با گسترش استفاده از ابزارهای تولید متن مبتنی بر هوش مصنوعی و ازدیاد داده های تقلبی، مسئله تشخیص متن تولید شده توسط هوش مصنویی اهمیت بالایی دارد، و این مسئله در بسیاری از زمینه‌ها از جمله آموزش، روزنامه‌نگاری، و تولید محتوا به یک چالش مهم تبدیل شده است. انجام یک تمرین طبقه‌بندی بر روی این داده‌ها می‌تواند به توسعه مدل‌هایی کمک کند که توانایی تشخیص متون انسان‌ساخته و هوش مصنوعی را بهبود بخشند، و این می‌تواند در جلوگیری از انتشار محتوای نادرست یا تولیدات غیرمعتبر بسیار موثر باشد.

به این منظور دیتاستی شامل اطلاعات متنی شامل حدود ۵۰۰ هزار مقاله، به طور خاص شامل متونی است که هم توسط انسان‌ها و هم توسط هوش مصنوعی تولید شده‌اند در اختیار شما قرار میگیرد، تا در ادامه به بررسی داده و تشخیص داده های تولید شده توسط هوش مصنوئی بپردازید.

https://www.kaggle.com/datasets/shanegerami/ai-vs-human-text

A. گوگل کولب را برای کار با کتابخانه spark آماده کنید(به پیوست مراجعه نمایید).
B. مجموعه داده را تحت فرمت دیتاست اسپارک باز کنید و مشخص کنید ستون تارگت شامل چه مقادیری است.
C. داده ها را به کمک اسپارک پیش پردازش کنید. به منظور هماهنگی نتایج مراحل زیر را انجام دهید:
   i. تمام کلمات را به حروف کوچک دربیاورید و کارکترهایی که خارج از دایره حروف الفبا و اعداد هستند را کنار بگذارید.
   ii. متن هر سطر را به صورت کلمه به کلمه تجزیه کنید. (توکنایز کنید)
   iii. کلمات اضافی (stop words) را از بین کلماتی که پیدا کردید حذف نمایید.
   (راهنمایی: از لینک زیر برای حذف کلمات اضافی استفاده کنید.
   https://spark.apache.org/docs/latest/api/python/reference/api/pyspark.ml.feature.StopWordsRemover.html
   در صورت نیاز لیست کلمات اضافی را مشابه لینک زیر از کتابخانه NLTK دریافت نمایید:
   (https://stackoverflow.com/questions/70698947/how-to-import-and-use-stopwords-list-from-nltk)
   iv. هر کلمه(توکن) را در یک سطر جداگانه قرار دهید. (پیشنهاد میشود، یک ستون جدید برای خروجی این بخش در نظر بگیرید.)
   (راهنمایی: https://hypercodelab.com/docs/spark/basics/explode-array-with-index)
D. هر کلمه با تعداد تکرارش را نمایش دهید. (نمایش ۲۰ کلمه کافیست)
E. کلمات با بیشترین تکرار را نمایش دهید. سپس کلمات با کمترین تکرار را نمایش دهید(نمایش ۱۰ کلمه از هرکدام کافیست).
F. کلمات منحصر به فرد (بدون تکرار) را نمایش دهید(۲۰ کلمه کافیست). تعداد کل کلمات منحصر به فرد را بیابید.
G. اکنون کلمات با بیشترین تکرار را برای هر کلاس به صورت جداگانه پیدا کنید و ده تا از آنها را نمایش دهید. آیا بین بیشینه کلمات استفاده شده توسط انسان و هوش‌مصنویی تفاوتی وجود دارد؟ بنظر شما این تفاوت در تشخیص تقلب کمک کننده است؟
H. با تغییر تعداد هسته ها حداقل تا ۸ هسته مجدد کد قسمت قبل (یافتن کلمات با بیشترین تکرار را برای هر کلاس) را مجدد اجرا نموده و زمان اجرا را برای تعداد هسته مختلف بدست آورده. نمودار زمان بر حسب تعداد هسته ها را رسم نمایید. آیا این زمانها و روند علت خاصی دارد؟

# سوال ۲)

اکنون که توانستیم با بررسی لغات مورد استفاده توسط انسان و هوش‌مصنوعی یا حدی کمی بین این دادگان مرزبندی کنیم، در ادامه تلاش می‌کنیم تا با به کار گیری هوش مصنوعی به نحو مناسب تری به حل این مسئله بپردازیم.

در این بخش نیز از دیتاست سوال ۱ استفاده نمایید.

A. مجموعه داده را تحت فرمت دیتاست اسپارک باز کنید و به کمک ابزارهای اسپارک پیش پردازش نمایید.
   i. تمام کلمات را به شکل کوچک در بیاورید و کارکترهایی که خارج از دایره حروف الفبا انگلیسی و اعداد هستند را کنار بگذارید.
   ii. متن هر سطر را یه صورت کلمه به کلمه تجزیه کنید. (توکنایز کنید)
   iii. کلمات اضافی (stop words) را از بین کلماتی که پیدا کردید حذف نمایید.
   (راهنمایی: از لینک زیر برای حذف کلمات اضافی استفاده کنید.
   https://spark.apache.org/docs/latest/api/python/reference/api/pyspark.ml.feature.StopWordsRemover.html
   می‌توانید لیست کلمات اضافی را مشابه لینک زیر از کتابخانه NLTK دریافت نمایید:
   (https://stackoverflow.com/questions/70698947/how-to-import-and-use-stopwords-list-from-nltk
   iv. در این مرحله داده را وکتورایز نمایید. تا به عنوان ورودی مدل به شکل مناسب تبدیل شوند.
   (وکتورایزر پیشنهادی :
   https://spark.apache.org/docs/latest/api/python/reference/api/pyspark.ml.feature.CountVectorizer.html )
   v. برای آموزش و ارزیابی مدل، داده را به صورت رندم با نسبت ۸۰ به ۲۰ به دو قسمت train و test تقسیم کنید. (به منظور هماهنگی نتایج مقدار seed را ۱۲۳ در نظر بگیرید)

B. در مورد روش یادگیری ماشینی Bayesian Classifier و Naive Bayes تحقیق و مختصرعملکرد آن‌ها را توضیح دهید. کدام برای انجام پروژه بیگدیتا مناسب تر است؟ چرا؟

C. یک مدل Naive Bayes از پکیج یادگیری ماشین اسپارک ایجاد کرده و دادگان را آموزش دهید.

D. داده های تست را برای پیشبینی به مدل بدهید و معیارهای ارزیابی طبقه بندی (Precision, acc, Recall و f1) را بدست بیاورید. (راهنمایی:
https://spark.apache.org/docs/latest/api/python/reference/api/pyspark.ml.evaluation.MulticlassClassificationEvaluator.html )

# سوال ۳)

خوشه بندی از جمله زیرمجموعه‌های مهم یادگیری ماشین بدون ناظر است، که در پردازش دادگان حجیم نیز بسیار مهم و کاربردی است. در این سوال میخواهیم به کمک PySpark خوشه‌بندی انجام دهیم. برای این کار دو دیتاست مختصات دادگان دوبعدی (با بیش از ۱ میلیون نمونه) با نام‌های dataset_clustring_1.csv و dataset_clustring_2.csv در اختیار شما قرار میگیرد. البته برای تحلیل بهتر کلاس نقاط هم در دیتاستها قرار دارد. برای هردو دیتاست مراحل زیر را جداگانه انجام دهید:

A. ابتدا نقاط را رسم کنید. برای اینکار از رنگ بندی مبتنی بر کلاسهای همراه نقاط استفاده کنید.
(راهنمایی: به علت ازدیاد نقاط و برای نمایش بهتر، مقدار پارامتر s را در pyplot.scatter مقدار کوچکی مانند ۱ در نظر بگیرید. https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.scatter.html )

B. درمورد روشهای K-Means و K-Means++ تحقیق کرده و به اختصار و کوتاه آن‌ها را توضیح دهید.

C. چگونه می‌توان از K-Means++ در PySpark استفاده نمود؟
به کمک الگوریتم K-Means++ نقاط را خوشه بندی کنید.
برای اینکار تعداد خوشه‌های ۲ تا ۲۵ را امتحان کنید.
(راهنمایی: https://spark.apache.org/docs/latest/api/python/reference/api/pyspark.ml.clustering.KMeans.html
در PySpark باید داده‌ها را با فقط یک ستون از نوع وکتور به به عنوان فیچر وکتور مدلهای ماشین لرنینگی دهیم.
https://spark.apache.org/docs/latest/api/python/reference/api/pyspark.ml.feature.VectorAssembler.html )

D. نمودار هزینه بر حسب تعداد خوشه‌ها را رسم کنید.
(راهنمایی: از ارزیابی‌گر زیر و معیار silhouette استفاده نمایید:
https://spark.apache.org/docs/latest/api/python/reference/api/pyspark.ml.evaluation.ClusteringEvaluator.html )

E. بهینه ترین تعداد خوشه را مشخص کنید.
(راهنمایی: برای اینکار میتوانید از الگوریتم‌هایی مانند KneeLocator (مستقل از pyspark) استفاده نمایید.
https://github.com/arvkevi/kneed )

F. با توجه به تعداد کلاستر بهینه، شماره کلاستر نقاط دیتاست را مشخص کرده و مراکز کلاسترها را پیدا کنید.

G. داده های کلاستر شده را رسم نمایید. رنگبندی نقاط را نیز براساس شماره کلاسترها انجام دهید. (بدون توجه به شماره کلاسها). (راهنمایی: مشابه راهنمایی بخش A)

H. نتایج رسم نقاط باشماره کلاسها(بخشA) و رسم نقاط با شماره کلاسترها(بخشG) را با هم مقایسه کنید. تعداد کلاسها واقعی دیتاست چندتا بود و تعداد کلاسترها یافته شده چندتا است؟ آیا K-Means++ در خوشه‌بندی هر دو دیتاست موفق بوده؟ به طور مختصر علت را توضیح دهید.

# سوال ۴)

- یکی از ابزارتی که به منظور تسهیل عملیات نگاشت کاهش (MapReduce) فراهم شده است، کتابخانه پایتونی MRjob است، که مشکلات و مسائل مربوط به مدریت کلاسترها را از دوش کاربر برداشته و فرآیند نگاشت و کاهش را آسان تر میکند.

A. به کمک کتابخانه MRjob برنامه ای بنویسید که تعداد تکرار کلمات متن را به روش MapReduce محاسبه کند. به این منظور دیتاست شامل متونی پیش‌پردازش شده از داستانهای هری‌پاتر با نام harrypotter_processed.txt در اختیار شما قرار میگیرد.
بخش‌های اصلی کد را در گزارش آورده و مختصر توضیح دهید که از لحاظ تئوری هر کدام در این پروسه چه کاری انجام می‌دهند. (راهنمایی: https://www.google.com/)

B. به کمک کتابخانه MRjob و به روش MapReduce میانگین دما هر ماه شهر تهران را که به صورت روزانه ذخیره شده است، محاسبه کنید. دیتاست بانام Tehran_temperature_data.txt در اختیار شما قرار میگیرد.
بخش‌های اصلی کد را در گزارش آورده و مختصر توضیح دهید که از لحاظ تئوری هر کدام در این پروسه چه کاری انجام می‌دهند. (راهنمایی میتوانید برای اینکار از گیتهاب زیر استفاده نمایید.

(https://gist.github.com/tmrts/5d21ffcafef19ba044ff5c34134b814d

- در این بخش از سوال میخواهیم به روش MapReduce و با استفاده از spark به انجام دو عملیات ماتریسی بپردازیم.

C. برنامه‌ای بنویسید که با استفاده از MapReduce، عملیات ضرب ماتریسی را انجام دهد.

(از مخزن گیت هاب الگو برداری کنید و در اسپارک پیادهسازی را انجام دهید.

(https://github.com/awasthishubh/mapReduce-matrix_mul

حاصل عملیات‌های زیر را با کد خود بدست آورید.

$$\begin{bmatrix}1 & 2 & 3\\ 4 & 5 & 6\end{bmatrix}\begin{bmatrix}7 & 8\\ 9 & 10\\ 11 & 12\end{bmatrix} \quad , \quad \begin{bmatrix}10 & 11\\ 12 & 13\end{bmatrix}\begin{bmatrix}14 & 15\\ 16 & 17\end{bmatrix}$$

D. برنامه‌ای بنویسید که با استفاده از MapReduce، عملیات جمع ماتریسی را انجام دهد.

حاصل عملیات‌های زیر را با کد خود بدست آورید.

$$\begin{bmatrix}0 & 0 & 0\\ 1 & 1 & 1\\ 2 & 2 & 2\end{bmatrix}+\begin{bmatrix}3 & 3 & 3\\ 4 & 4 & 4\\ 5 & 5 & 5\end{bmatrix} \quad , \quad \begin{bmatrix}10 & 11\\ 12 & 13\end{bmatrix}+\begin{bmatrix}14 & 15\\ 16 & 17\end{bmatrix}$$

در هر قسمت(Bو C) :

- بخش‌های اصلی کد را در گزارش آورده و مختصر توضیح دهید که از لحاظ تئوری هر کدام در این پروسه چه کاری انجام می‌دهند.
- الگوریتم خود را طوری پیاده‌سازی نمایید که قابل استفاده برای سایر ماتریس‌ها با ابعاد متفاوت هم باشد.

# پیوست

۱- اگر در اتصال به GoogleColab مشکل دارید، ابزارهای کمکی را امتحان کنید:

https://shecan.ir/tutorials/tutorials-free/
https://www.yasdl.com/273157/%D8%AF%D8%A7%D9%86%D9%84%D9%88%D8%AF-403-online.html

۲- کد اتصال GoogleDrive به GoogleColab :

```
from google.colab import drive
drive.mount('/content/drive')
```

۲- کد راه‌اندازی PySpark روی GoogleColab :

```
#  java installion
!apt-get install openjdk-17-jdk-headless -qq > /dev/null

# install spark (note! : check version number)
!wget -q https://archive.apache.org/dist/spark/spark-3.5.0/spark-3.5.0-bin-hadoop3.tgz
# unzip the spark file to the current folder
!tar xf spark-3.5.0-bin-hadoop3.tgz

# path of spark folder
import os
os.environ["JAVA_HOME"] = "/usr/lib/jvm/java-17-openjdk-amd64"
os.environ["SPARK_HOME"] = "/content/spark-3.5.0-bin-hadoop3"
os.environ["PATH"] = os.environ["SPARK_HOME"] + "/bin:" + os.environ["PATH"]
```

```
# install findspark
!pip install -q findspark
!pip install pyspark
```

```
# initialize and find findspark
import findspark
findspark.init()
findspark.find()
```

# موارد تحویلی

۱- برنامه های نوشته شده برای هر سوال

۲- گزارش تکلیف انجام شده که حداقل باید شامل : توضیح سوالات، ایده و الگوریتم طراحی شده برای حل مسائل، توضیح برنامه های نوشته شده، خروجی های خواسته شده برای سوالات و تحلیل نتایج بدست آمده باشد.

۳- موارد ذکر شده را در قالب یک فایل زیپ zip. در سامانه‌ی کورسز بارگذاری نمایید

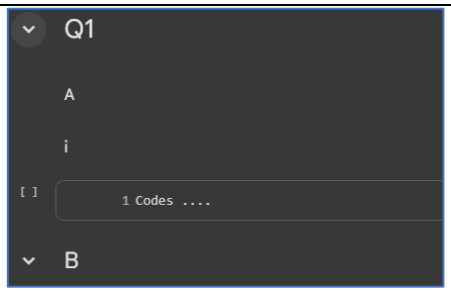
## توجه :

۱- لطفا کدنویسی پروژه خود را در قالب نوتبوک ipynb (ترجیحا در google colab) انجام دهید.

۲- کدها باید کامنت گذاری شده و تا حد امکان خوانا و تمیز باشند.

۳- در ابتدا گزارش فهرست قرار دهید. هدربندی ها را مشابه بخش‌بندی سوالات انجام دهید.

برای مثال برای کدنویسی زیر بخشiاز بخشAدر سوال 1 :

| در گزارش | در نوتبوک |
| --- | --- |
|  |  |

۴- بجز در موارد خواسته شده، نیازی به توضیح دقیق و خط به خط کد ها نیست، و تنها توضیح کارکرد کلی و برنامه و بلوکهای آن، کافیست.

۵- برای پاسخگویی به سوالات و تهیه گزارش، حتما خروجی هر بخش خواسته شده و یا اسکرین شات لاگ و نتایج مرتبط با سوال را همراه توضیح آن بخش ارائه دهید.

۶- برای کسب نمره کامل، توضیح هر بخش باید شفاف و کافی باشد. لطفا از توضیحات خیلی طولانی و ارائه مطالب حاشیه‌ای و کم اهمیت اجتناب نمایید. چراکه ممکن است، به علت پراکندی و حجم زیاد متن گزارش، مطالب اصلی و صحیح ارائه شده، از نگاه مصحح پنهان بمانند.

با سپاس و آرزوی پیروزمندی