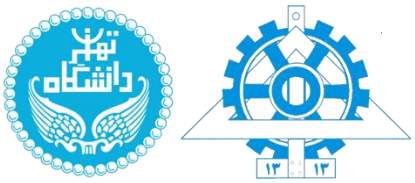
به نام خدا



سامانه‌های یادگیری ماشین توزیع‌شده (پاییز ۱۴۰۲)

تمرین کامپیوتری ۱

موعد تحویل: ۱۴۰۲/۸/۳

لطفا پیش از شروع کار بر روی تمارین، به نکات زیر توجه فرمایید.

- حتما ویدئوی راه‌اندازی کلاستر را به دقت مشاهده کنید و مطمئن شوید به کلاستر درس دسترسی دارید.
- برای راحتی در توسعه و تست کد، از ماشین مجازی لینوکس خود استفاده نمایید تا ترافیک کلاستر (به خصوص درساعات آخر مهلت تمرین) افزایش نیابد. پس از اطمینان از عملکرد کد، میتوانید آن را روی کلاستر اجرا کنید.
- توصیه میشود از زبان Python به همراه با کتابخانه mpi4py برای کدنویسی استفاده کنید. در آدرس زیر، راهنمای استفاده از این کتابخانه آمده است:

https://mpi4py.readthedocs.io/en/stable/tutorial.html

- سوالات خود را در گروه تلگرام درس مطرح نمایید. به هیچ وجه کد یا پاسخ سوالات را در گروه به اشتراک نگذارید.
- میزان تاخیر، تا دو روز مجاز است. تاخیر به صورت ساعتی محاسبه شده و هر روز ۱۰ درصد نمره کم میشود. تحویل تمرین پس از دو روز تاخیر امکانپذیر نخواهد بود.

۱. (۴۰ نمره) در این تمرین قصد داریم عدد اویلر (e) را به کمک روش آماری مونت کارلو محاسبه کنیم. در این روش، در هر گام، آن قدر عدد تصادفی از توزیع یکنواخت ۰ تا ۱ تولید می‌شود تا مجموعشان بیشتر از ۱ شود. برای تخمین عدد اویلر، کافیست میانگین تعداد دفعاتی که در هر گام اعداد تصادفی تولید شده بودند، محاسبه شود [1]. در این تمرین، با فرض 4,000,000 گام، این الگوریتم را در شرایط زیر پیاده سازی نمایید:

الف) (۵ نمره) کد سریال را بنویسید و بر روی ۱ نود و ۱ هسته اجرا نمایید.

ب) (۱۲ نمره) به کمک کتابخانه‌ی mpi4py ، کد قسمت (الف) را به صورت موازی سازی پیاده سازی کرده و بر روی ۱ نود و ۲ هسته اجرا نمایید.

ج) (۸ نمره) کد بخش (ب) را بر روی ۲ نود و ۲ هسته اجرا نمایید.

(۱۵ نمره) در هر آزمایش، نتیجه‌ی تخمین و زمان اجرایی را گزارش کرده و نتایج بخش‌های مختلف را با هم مقایسه کرده و تحلیل نمایید.

۲. (۶۰ نمره) یکی از کاربردهای الگوریتم‌های توزیع شده، حفظ حریم شخصی کاربران است. برای پیاده سازی این الگوریتم‌ها، باید در نظر گرفت که هر نود داده‌های مخصوص به خود را دارد و تمام یا بخش‌هایی از الگوریتم را خودش اجرا می‌کند و تنها برای تجمیع نتایج، نودها با یکدیگر ارتباط برقرار می‌کنند. به طور کلی، برای تقسیم داده‌ها بین نودها دو روش تقسیم بندی افقی و عمودی مطرح می‌شود. در تقسیم بندی افقی، هر نود تعدادی از رکوردهای داده را می‌گیرد و هر رکورد تمام ویژگی‌هایش را دارد. در حالی که در

تقسیم بندی عمودی، تقسیم بندی بر اساس ویژگی‌های داده‌ها انجام می‌شود و هر نود تعدادی از ویژگی‌ها را برای تمام رکوردها را دارد [2].

در این تمرین قصد داریم الگوریتم خوشه بندی K-Means را به صورت توزیع شده پیاده سازی کنیم. برای رعایت حفظ حریم شخصی، قصد داریم از روش تقسیم بندی افقی استفاده نماییم. به این منظور، الگوریتم K-Means را به صورت زیر تغییر می‌دهیم:

۱) نود اصلی مراکز را به صورت رندوم تولید و به نودهای دیگر ارسال می‌کند (برای این کار می‌توانید فرض کنید نود اصلی دامنه‌ی تغییرات داده‌ها دارد).
۲) هر نود، با توجه به مراکز دریافتی، خوشه بندی را برای داده‌هایش انجام می‌دهد.
۳) هر نود، فاصله‌ی داده‌هایش تا مرکز هر خوشه را به دست آورده و مجموع فاصله‌ی داده‌ها به همراه تعدادشان را به نود اصلی ارسال می‌کند.
۴) نود اصلی با دریافت مجموع فاصله‌ها و تعداد نقاط هر خوشه، مراکز خوشه‌ها را به روز رسانی کرده و آن‌ها را دوباره به نودهای دیگر ارسال می‌کند. سپس دوباره الگوریتم از گام ۲ اجرا می‌شود.

برای این سوال، از مجموعه داده‌ی data.csv استفاده نمایید و الگوریتم خوشه بندی را با ۵ خوشه و ۱۰ iteration اجرا نمایید. این الگوریتم را در شرایط زیر پیاده سازی کنید:

الف) (۱۰ نمره) پیاده سازی K-Means به صورت سریال را انجام دهید و آن را بر روی ۱ نود و ۱ هسته اجرا نمایید.

ب) (۱۸ نمره) به کمک کتابخانه‌ی mpi4py، کد قسمت (الف) را برای K-Means توزیع شده پیاده سازی نمایید و بر روی ۱ نود و ۲ هسته اجرا کنید.

ج) (۱۲ نمره) کد قسمت (ب) را بر روی ۲ نود و ۲ هسته اجرا نمایید.

(۲۰ نمره) در هر آزمایش، نمودار خوشه بندی نهایی و زمان اجرایی را گزارش کرده و نتایج بخش‌های مختلف را با هم مقایسه کرده و تحلیل نمایید.

۳. (امتیازی) (۱۵ نمره) در این تمرین قصد بنچمارک کردن پیاده سازی‌های مختلف BLAS در کتابخانه‌ی numpy را داریم. برای نصب کتابخانه‌های شتاب‌دهنده‌ی جبر خطی مختلف به همراه numpy، می‌توانید راهنمای نصب numpy را مطالعه نمایید. برای بنچمارک کردن این کتابخانه‌ها، می‌توانید عملیات‌های ضرب ماتریسی، محاسبه مقدار ویژه و بردار ویژه و محاسبه رگرسیون OLS را برای داده‌هایی با سایزهای مختلف انجام دهید. همچنین برای بدست آوردن زمان و حافظه‌ی مصرفی می‌توانید هم از ابزارهای پروفایلر لینوکس (مانند /usr/bin/time) و هم ابزارهای پروفایلر پایتون استفاده نمایید. در نهایت، آزمایش‌ها را بر روی پیاده سازی‌های مختلف BLAS انجام داده و با یکدیگر مقایسه نمایید (برای این سوال از کامپیوتر شخصیتان استفاده نمایید).

(کد: ۱۰ نمره + گزارش: ۵ نمره) نتایج آزمایش‌های مختلفی که انجام داده‌اید را گزارش کرده و با یکدیگر مقایسه نمایید.

نحوه تحویل پروژه

فایل‌ها را به صورت زیر نام گذاری کرده و در آخر همه را در یک فایل zip در سامانه ارسال کنید:

۱- گزارش report.pdf

۲- نام گذاری کدها را به صورت زیر انجام دهید:

سوال	بخش	نام فایل
۱	الف	e_sim_a.py e_sim_a.sh
	ب	e_sim_b.py e_sim_b.sh
	ج	e_sim_c.py e_sim_c.sh
۲	الف	k_means_a.py k_means_a.sh
	ب	k_means_a.py k_means_a.sh
	ج	k_means_a.py k_means_a.sh
۳	_	فایل کدهایی که زده‌اید را با پیشوند _bench نام گذاری نمایید.

[1] Russel KG. 1991. Estimating the Value of e by Simulation. In *The American Statistician, Vol. 45, No. 1* , pages 66-68

[2] Aggarwal CC, Philip SY (Eds.). 2008. Privacy-preserving data mining: models and algorithms. In *Springer Science and Business Media*. page 28