**Bài thực hành số 4**

**Phân tích dữ liệu điểm thi THPT Quốc gia bằng**

**RDD**

**Link bài làm trên Databricks:** <https://databricks-prod-cloudfront.cloud.databricks.com/public/4027ec902e239c93eaaa8714f173bcfc/345358637122346/4132740522268171/5017236833160125/latest.html>

**Bài làm:**

def fsplit(s):

    l = s.split(',')

    d = {'TT':int(l[0]), 'SBD':l[11]}

    if l[1]!='':

        d['Dia'] = float(l[1])

    if l[2]!='':

        d['GDCD'] = float(l[2])

    if l[3]!='':

        d['Hoa'] = float(l[3])

    if l[4]!='':

        d['Li'] = float(l[4])

    if l[6]!='':

        d['NN'] = float(l[6])

    if l[7]!='':

        d['Sinh'] = float(l[7])

    if l[8]!='':

        d['Su'] = float(l[8])

    if l[9]!='':

        d['Toan'] = float(l[9])

    if l[10]!='':

        d['Van'] = float(l[10])

    return [d]

rdd = sc.textFile('/FileStore/Bigdata\_task04/diemthi2020.csv').filter(lambda s: s[0]!='T') .flatMap(fsplit)

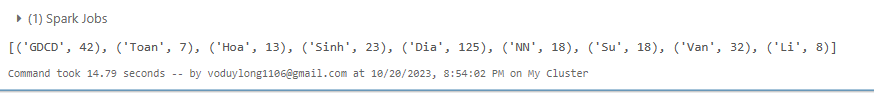
**1. Đếm số bài thi điểm 0 từng môn.**

rdd1 = rdd.flatMap(lambda h: [(x, y) for x, y in h.items()]).\

    filter(lambda t: not (t[0] in ['TT', 'SBD'])).groupByKey().\

    mapValues(lambda scores: sum(1 for s in scores if s == 0))

print(rdd1.collect())

****

**2. Cho biết môn thi có nhiều bài thi đạt điểm 10 nhất và số bài thi điểm 10 môn đó là bao nhiêu.**

rdd2 = rdd.flatMap(lambda h: [(x, y) for x, y in h.items()]).\

    filter(lambda t: not (t[0] in ['TT', 'SBD'])).groupByKey().\

    mapValues(lambda scores: sum(1 for s in scores if s == 10))

result2 = rdd2.reduce(lambda x, y: x if x[1] > y[1] else y)

print(result2)

****

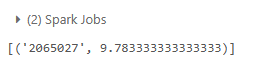
**3. Cho biết số báo danh của những thí sinh có điểm trung bình các môn thi cao nhất.**

rdd3 = rdd.map(lambda t: (t['SBD'], sum(t[i] for i in t if i != 'TT' and i != 'SBD') / (len(t) - 2)))

max\_avg\_score = rdd3.reduce(lambda x, y: x if x[1] > y[1] else y)

list\_max\_avg\_score = rdd3.filter(lambda x: x[1] == max\_avg\_score[1])

print(list\_max\_avg\_score.collect())

****

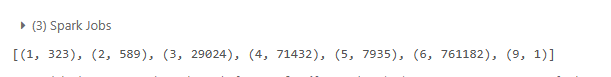
**4. Thống kê số lượng thí sinh theo số môn thi. Ví dụ: 1 môn có bao nhiêu thí sinh, 2 môn .., ..., 9 môn có bao nhiêu thí sinh.**

rdd4 = rdd.map(lambda t: (sum(1 for i in t if i != 'TT' and i != 'SBD'), t['SBD'])).\

    groupByKey().mapValues(len).\

    sortBy(lambda x: x[0])

print(rdd4.collect())

****

**5. Tính điểm trung bình bài thi Tự nhiên, bài thi Xã hội. Biết rằng điểm bài thi tổ hợp là tổng điểm thi 3 môn thành phần của bài thi đó.**

def trung\_binh\_tu\_nhien(sv):

    return round(((sv['Li'] + sv['Hoa'] + sv['Sinh']) / 3), 2)

def trung\_binh\_xa\_hoi(sv):

    return round(((sv['Su'] + sv['Dia'] + sv['GDCD']) / 3), 2)

rdd5\_tn = rdd.filter(lambda x: 'Li' in x and 'Hoa' in x and 'Sinh' in x)

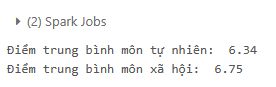
result\_tn = rdd5\_tn.map(lambda x: (trung\_binh\_tu\_nhien(x))).mean()

rdd5\_xh = rdd.filter(lambda x: 'Su' in x and 'Dia' in x and 'GDCD' in x)

result\_xh = rdd5\_xh.map(lambda x: (trung\_binh\_xa\_hoi(x))).mean()

print("Điểm trung bình môn tự nhiên: ", round(result\_tn,2))

print("Điểm trung bình môn xã hội: ", round(result\_xh, 2))

****

**6. Vẽ biểu đồ biểu diễn số lượng bài thi Tự nhiên và Xã hội theo từng mức điểm.**

def tong\_diem\_tn(sv):

    return round((sv['Li'] + sv['Hoa'] + sv['Sinh']), 0)

def tong\_diem\_xh(sv):

    return round(sv['Su'] + sv['Dia'] + sv['GDCD'], 0)

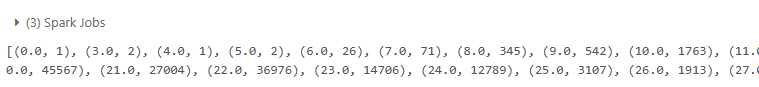
rdd6\_tn = rdd5\_tn.map(lambda x: {'SBD': x['SBD'], 'Hoa': x['Hoa'], 'Li': x['Li'], 'Sinh': x['Sinh']}).\

    map(lambda y: (tong\_diem\_tn(y), y['SBD'])).\

    groupByKey().mapValues(len).\

    sortBy(lambda z: z[0])

print(rdd6\_tn.collect())

****

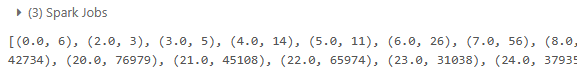
rdd6\_xh = rdd5\_xh.map(lambda x: {'SBD': x['SBD'], 'Su': x['Su'], 'Dia': x['Dia'], 'GDCD': x['GDCD']}).\

    map(lambda y: (tong\_diem\_xh(y), y['SBD'])).\

    groupByKey().mapValues(len).\

    sortBy(lambda z: z[0])

print(rdd6\_xh.collect())

****

import matplotlib.pyplot as plt

fig,a= plt.subplots(2,1, figsize=(20,10))

x\_axis = list(range(1, 31))

x1 = [i[0] for i in rdd6\_tn.collect()]

y1 = [i[1] for i in rdd6\_tn.collect()]

x2 = [i[0] for i in rdd6\_xh.collect()]

y2 = [i[1] for i in rdd6\_xh.collect()]

a[0].bar(x1, y1, 0.4, color='blue')

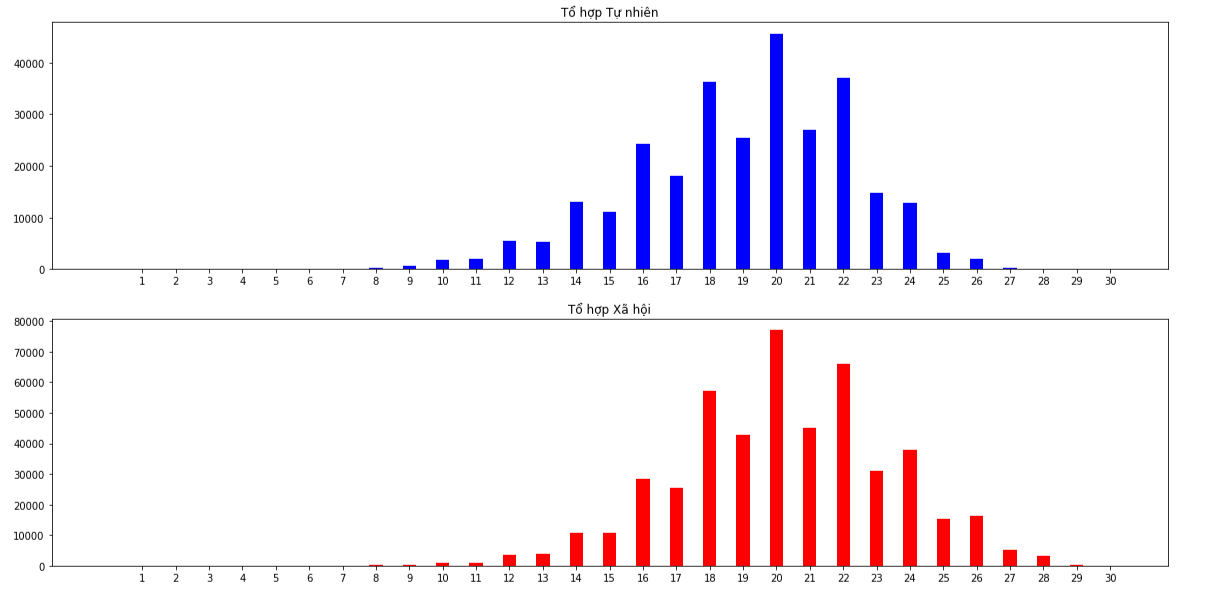
a[0].set\_xticks(x\_axis)

a[0].set\_title('Tổ hợp Tự nhiên')

a[1].bar(x2, y2, 0.4, color='red')

a[1].set\_xticks(x\_axis)

a[1].set\_title('Tổ hợp Xã hội');

****

**7. Vẽ biểu đồ boxplot thể hiện thống kê điểm các môn: Toán, Văn, Ngoại ngữ.**

import seaborn

rdd7\_toan = rdd.filter(lambda x: 'Toan' in x).\

    map(lambda y: ('Toan', y['Toan'])).groupByKey()

rdd7\_van = rdd.filter(lambda x: 'Van' in x).\

    map(lambda y: ('Van', y['Van'])).groupByKey()

rdd7\_nn = rdd.filter(lambda x: 'NN' in x).\

    map(lambda y: ('NN', y['NN'])).groupByKey()

toan\_score = rdd7\_toan.map(lambda x: list(x[1])).collect()

van\_score = rdd7\_van.map(lambda x: list(x[1])).collect()

nn\_score = rdd7\_nn.map(lambda x: list(x[1])).collect()

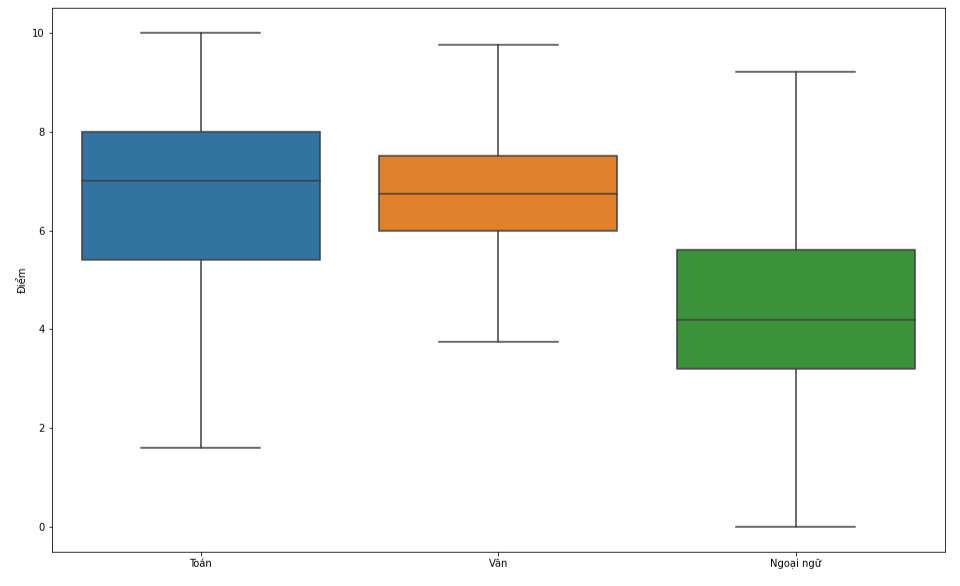
plt.figure(figsize=(16, 10))

seaborn.boxplot(data=[toan\_score, van\_score, nn\_score], showfliers=False)

plt.xticks([0, 1, 2], ['Toán', 'Văn', 'Ngoại ngữ'])

plt.ylabel('Điểm')

plt.show()

****

**8. Cho biết mã tỉnh có nhiều thí sinh có điểm thi 10. Biết rằng mỗi tỉnh khác nhau ở 2 ký tự đầu trong số báo danh của thí sinh.**

rdd8 = rdd.map(lambda d: (d['SBD'][:2], d)).\

    map(lambda record: (record[0], sum(1 for key, score in record[1].items() if key not in ['TT', 'SBD'] and score == 10))).\

    groupByKey().mapValues(sum).\

    sortBy(lambda z: z[1], ascending=False)

print(rdd8.take(1))

****

**9. Thống kê điểm trung bình theo từng tỉnh, vẽ biểu đồ cho thống kê này.**

def fmean(l):

    return round((sum(l)/len(l)), 2)

rdd9 = rdd.map(lambda d: (d['SBD'][:2], d)).\

    map(lambda record: (record[0], sum(score for key, score in record[1].items() if key not in ['TT', 'SBD']))).\

    groupByKey().mapValues(fmean).\

    sortBy(lambda z: z[0])

print(rdd9.collect())

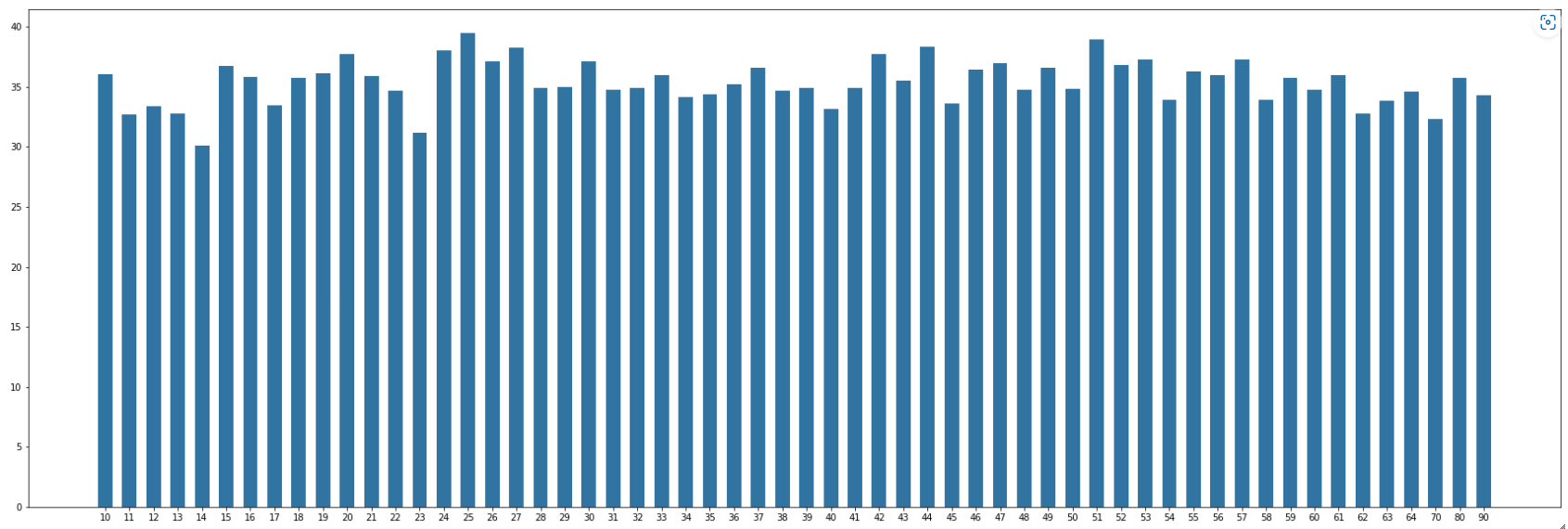
plt.figure(figsize=(30, 10))

x = [i[0] for i in rdd9.collect()]

y = [i[1] for i in rdd9.collect()]

plt.bar(x, y, 0.6, color='#3274a1')

plt.show()

****

**10.Thống kê mức điểm theo khối A (Toán, Lí, Hóa), B (Toán, Hóa, Sinh), C(Văn, Sử, Địa), D(Toán, Văn, NN). Mỗi thí sinh chỉ lấy 1 khối có tổng điểm cao nhất.**

def khoi\_diem\_cao\_nhat(sv):

    d = {}

    if 'Toan' in sv and 'Li' in sv and 'Hoa' in sv:

        d['A'] = round((sv['Toan'] + sv['Li'] + sv['Hoa']), 1)

    if 'Toan' in sv and 'Sinh' in sv and 'Hoa' in sv:

        d['B'] = round((sv['Toan'] + sv['Sinh'] + sv['Hoa']), 1)

    if 'Van' in sv and 'Su' in sv and 'Dia' in sv:

        d['C'] = round((sv['Van'] + sv['Su'] + sv['Dia']), 1)

    if 'Toan' in sv and 'Van' in sv and 'NN' in sv:

        d['D'] = round((sv['Toan'] + sv['Van'] + sv['NN']), 1)

    if len(d) > 0:

        return max(d.items(), key=lambda item: item[1])

rdd10 = rdd.flatMap(lambda x: [khoi\_diem\_cao\_nhat(x)]).\

    filter(lambda y: y is not None)

rdd10\_avg = rdd10.groupByKey().mapValues(fmean)

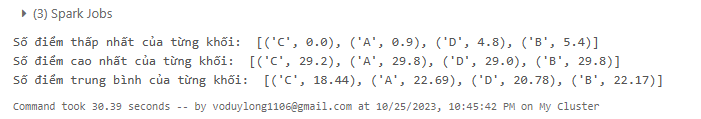
rdd10\_min = rdd10.groupByKey().mapValues(min)

rdd10\_max = rdd10.groupByKey().mapValues(max)

print('Số điểm thấp nhất của từng khối: ', rdd10\_min.collect())

print('Số điểm cao nhất của từng khối: ', rdd10\_max.collect())

print('Số điểm trung bình của từng khối: ', rdd10\_avg.collect())

****

**11. Vẽ biểu đồ cho thống kê mức điểm tổng theo khối.**

plt.figure(figsize=(30, 10))

x = [i[0] for i in rdd10.groupByKey().mapValues(fmean).collect()]

y = [i[1] for i in rdd10.groupByKey().mapValues(fmean).collect()]

plt.bar(x, y, 0.6, color='#3274a1')

plt.ylabel('Điểm trung bình', fontsize=16)

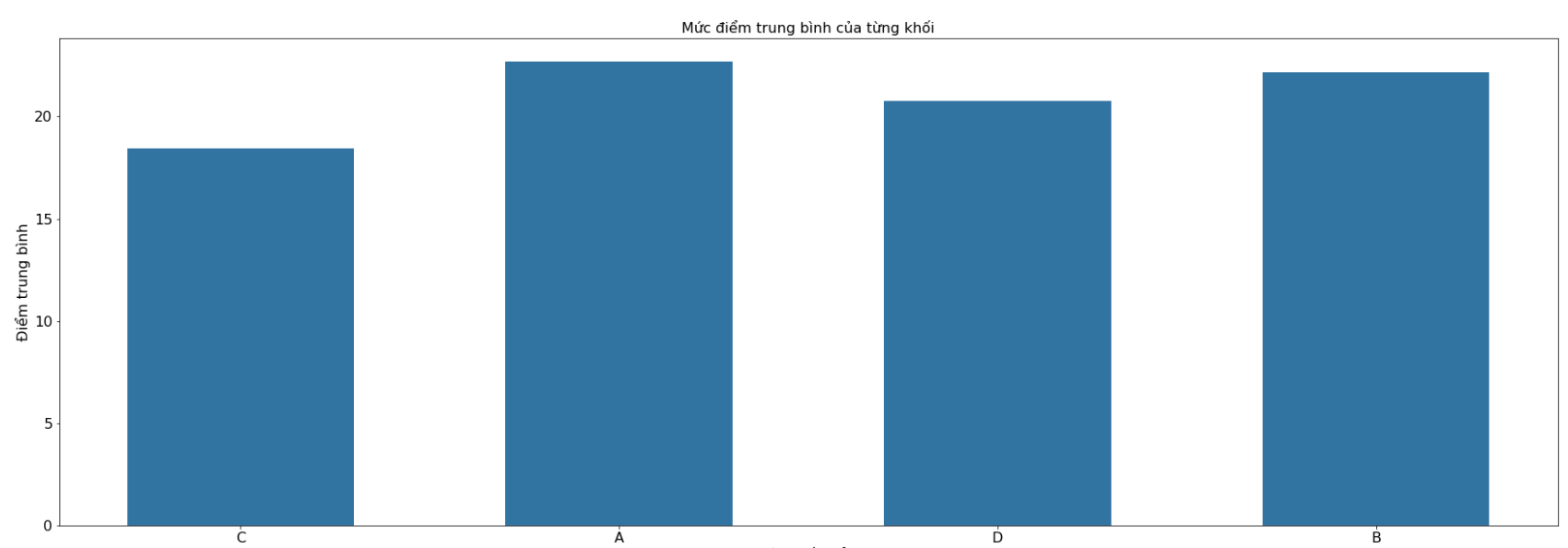
plt.xlabel('Các khối điểm', fontsize=16)

plt.xticks(fontsize=16)

plt.yticks(fontsize=16)

plt.title('Mức điểm trung bình của từng khối', fontsize=16)

plt.show()

****