main

July 25, 2021

1 Summary of prediction analysis for DE genes

[1]:
```
import os, errno
import pandas as pd
import seaborn as sns
from venn import venn
import matplotlib.pyplot as plt
```

1.1 Functions

[2]:
```
def mkdir_p(directory):
    """
    Make a directory if it does not already exist.

    Input: Directory name
    """
    try:
        os.makedirs(directory)
    except OSError as e:
        if e.errno != errno.EEXIST:
            raise
```

1.2 Load and prep summary files

1.2.1 Load files

[3]:
```
rf0 = pd.read_csv("../../rf/summary_10Folds_allTissues.tsv", sep='\t')
enet0 = pd.read_csv("../../enet/summary_10Folds_allTissues.tsv", sep='\t')
degs = pd.read_csv("../../../../_m/degs_annotation.txt", sep='\t', index_col=0)
```

1.2.2 Group, select, and clean summary results

[4]:
```
## Extract median of model metrics over 10 folds
rf = rf0.groupby(["tissue", "feature"]).median()\
        .loc[:, ["n_features", "test_score_r2"]].reset_index()
rf.feature = rf.feature.str.replace("_", ".", regex=True)
rf["Model"] = "Random Forest"
enet = enet0.groupby(["tissue", "feature"]).median()\
```

```
        .loc[:, ["n_features", "test_score_r2"]].reset_index()
enet.feature = enet.feature.str.replace("_", ".", regex=True)
enet["Model"] = "Elastic Net"

df = pd.concat([rf, enet], axis=0)
df.head(2)
```

[4]:

	tissue	feature	n_features	test_score_r2	Model
0	Caudate	ENSG00000003249.13	33.5	0.037818	Random Forest
1	Caudate	ENSG00000003509.15	3.0	-0.063606	Random Forest

1.2.3 Add partial r2 results

[5]:
```
partial = pd.read_csv("../../partial_r2/rf_partial_r2_metrics.tsv", sep='\t')\
            .rename(columns={"Geneid": "Feature"})
partial.columns = partial.columns.str.lower()
partial["test_score_r2"] = partial.partial_r2
partial["Model"] = "Partial R2"
partial = partial.loc[:, ['tissue', 'feature', 'n_features', 'test_score_r2',␣
↪'Model']]
partial.head(2)
```

[5]:

	tissue	feature	n_features	test_score_r2	Model
0	Caudate	ENSG00000003249.13	33	0.243673	Partial R2
1	Caudate	ENSG00000003509.15	2	0.013140	Partial R2

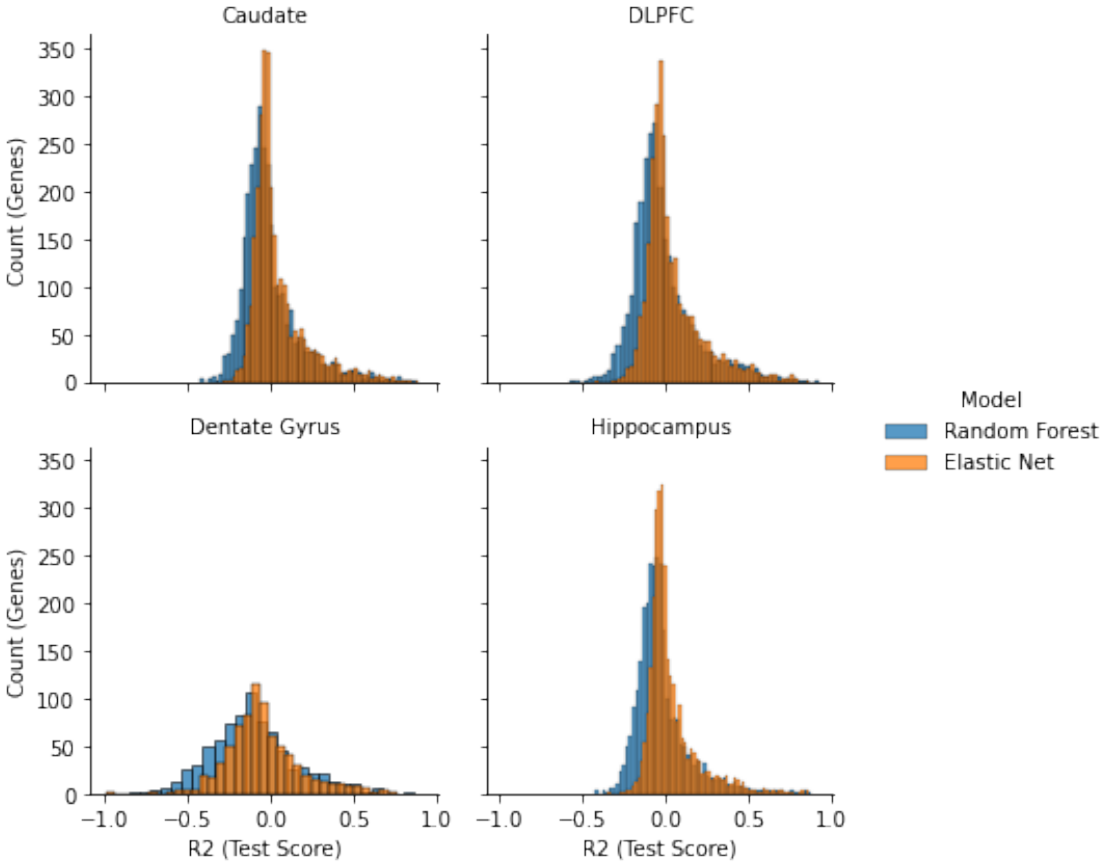
[6]:
```
df2 = pd.concat([df, partial], axis=0)
df2.groupby(["tissue", "Model"]).size()
```

[6]:
```
tissue         Model
Caudate        Elastic Net      2929
               Partial R2       2867
               Random Forest    2929
DLPFC          Elastic Net      2711
               Partial R2       2660
               Random Forest    2711
Dentate Gyrus  Elastic Net       773
               Partial R2        770
               Random Forest     773
Hippocampus    Elastic Net      2911
               Partial R2       2843
               Random Forest    2911
dtype: int64
```

1.3 Summary of results

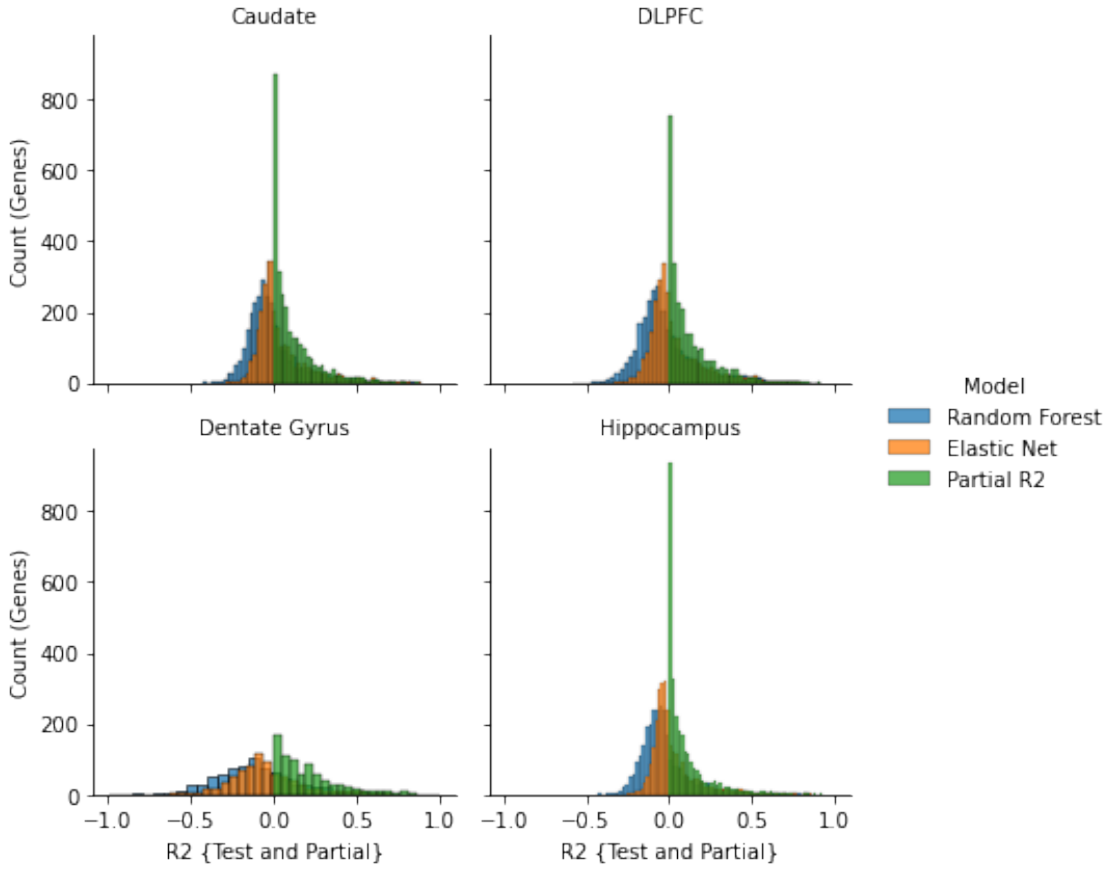
1.3.1 Histogram of R2 (median test R2 score)

[7]:
```
grid = sns.FacetGrid(df, col="tissue", col_wrap=2, hue="Model")
grid.map(sns.histplot, "test_score_r2")
grid.set_axis_labels("R2 (Test Score)", "Count (Genes)")
grid.set_titles(col_template="{col_name}")
grid.add_legend()
grid.tight_layout()
grid.savefig("histogram_test_r2.pdf")
grid.savefig("histogram_test_r2.png")
grid.savefig("histogram_test_r2.svg")
```



[8]:
```
grid = sns.FacetGrid(df2, col="tissue", col_wrap=2, hue="Model")
grid.map(sns.histplot, "test_score_r2")
grid.set_axis_labels("R2 {Test and Partial}", "Count (Genes)")
grid.set_titles(col_template="{col_name}")
grid.add_legend()
grid.tight_layout()
```

```
grid.savefig("histogram_test_N_partial_r2.pdf")
grid.savefig("histogram_test_N_partial_r2.png")
grid.savefig("histogram_test_N_partial_r2.svg")
```



1.3.2 What number of DEGs do not have any SNPs within 20 Kbp of gene body?

[9]:
```
for tissue in ["Caudate", "DLPFC", "Hippocampus", "Dentate Gyrus"]:
    xx = set(df[(df["tissue"] == tissue)].feature)
    yy = set(degs[(degs["Tissue"] == tissue)].index)
    txt = "{} of {} ({:.1%}) of DE genes do not have SNPs within 20Kbp."
    print(txt.format(len(yy) - len(xx), len(yy), (len(yy) - len(xx)) / len(yy)))
```

```
41 of 2970 (1.4%) of DE genes do not have SNPs within 20Kbp.
49 of 2760 (1.8%) of DE genes do not have SNPs within 20Kbp.
45 of 2956 (1.5%) of DE genes do not have SNPs within 20Kbp.
13 of 786 (1.7%) of DE genes do not have SNPs within 20Kbp.
```

1.3.3 Number of ancestry DE genes expression that can be predictive with SNP

[10]:
```
df[(df["test_score_r2"] >= 0.5)].groupby(["tissue", "Model"]).size()
```

[10]:
```
tissue         Model
Caudate        Elastic Net      92
               Random Forest    74
DLPFC          Elastic Net      92
               Random Forest    69
Dentate Gyrus  Elastic Net      20
               Random Forest    14
Hippocampus    Elastic Net      56
               Random Forest    52
dtype: int64
```

[11]:
```
df[(df["test_score_r2"] >= 0.75)].groupby(["tissue", "Model"]).size()
```

[11]:
```
tissue         Model
Caudate        Elastic Net      11
               Random Forest    11
DLPFC          Elastic Net      11
               Random Forest     7
Dentate Gyrus  Elastic Net       1
               Random Forest     1
Hippocampus    Elastic Net      12
               Random Forest    13
dtype: int64
```

[12]:
```
print(df[(df["test_score_r2"] >= 0.85)].groupby(["tissue", "Model"]).size().
 ↪reset_index())
df[(df["test_score_r2"] >= 0.85)]
```

```
          tissue          Model  0
0        Caudate    Elastic Net  2
1        Caudate  Random Forest  1
2          DLPFC    Elastic Net  1
3          DLPFC  Random Forest  2
4  Dentate Gyrus  Random Forest  1
5    Hippocampus    Elastic Net  2
6    Hippocampus  Random Forest  1
```

[12]:

	tissue	feature	n_features	test_score_r2	\
34	Caudate	ENSG00000013573.16	28.5	0.875303	
4219	DLPFC	ENSG00000166435.15	57.5	0.851186	
4936	DLPFC	ENSG00000226278.1	16.0	0.919475	
6205	Dentate Gyrus	ENSG00000226278.1	10.5	0.860346	
7765	Hippocampus	ENSG00000166435.15	15.0	0.871976	
34	Caudate	ENSG00000013573.16	36.0	0.890378	

```
1313        Caudate  ENSG00000166435.15        20.5       0.878129
4219          DLPFC  ENSG00000166435.15        19.5       0.864944
7765    Hippocampus  ENSG00000166435.15        19.5       0.863666
8952    Hippocampus   ENSG00000256274.1        26.0       0.861908

              Model
34    Random Forest
4219  Random Forest
4936  Random Forest
6205  Random Forest
7765  Random Forest
34      Elastic Net
1313    Elastic Net
4219    Elastic Net
7765    Elastic Net
8952    Elastic Net
```

[13]:
```
set(df[(df["test_score_r2"] >= 0.85)].feature)
```

[13]:
```
{'ENSG00000013573.16',
 'ENSG00000166435.15',
 'ENSG00000226278.1',
 'ENSG00000256274.1'}
```

- **ENSG00000166435.15** is *XRRA1* one of the most significant eQTLs in the brain
- **ENSG00000013573.16** is *DDX11*
- **ENSG00000226278.1** is *PSPHP1* a pseudogene
- **ENSG00000256274.1** is *TAS2R64P* anohter pseudogene

[14]:
```
print(df[(df["test_score_r2"] >= 0.9)].groupby(["tissue", "Model"]).size().
 ↪reset_index())
df[(df["test_score_r2"] >= 0.9)]
```

```
  tissue          Model  0
0  DLPFC  Random Forest  1
```

[14]:
```
      tissue            feature  n_features  test_score_r2          Model
4936  DLPFC  ENSG00000226278.1        16.0       0.919475  Random Forest
```

1.3.4 What is the overlap between models?

[15]:
```
for tissue in ["Caudate", "DLPFC", "Hippocampus", "Dentate Gyrus"]:
    print(tissue)
    for r2 in [0, 0.2, 0.5, 0.6, 0.7, 0.75, 0.8, 0.825]:
        ee = enet[(enet["tissue"] == tissue) & (enet["test_score_r2"] >= r2)].
 ↪copy()
        rr = rf[(rf["tissue"] == tissue) & (rf["test_score_r2"] >= r2)].copy()
        oo = len(set(ee.feature) & set(rr.feature))
```

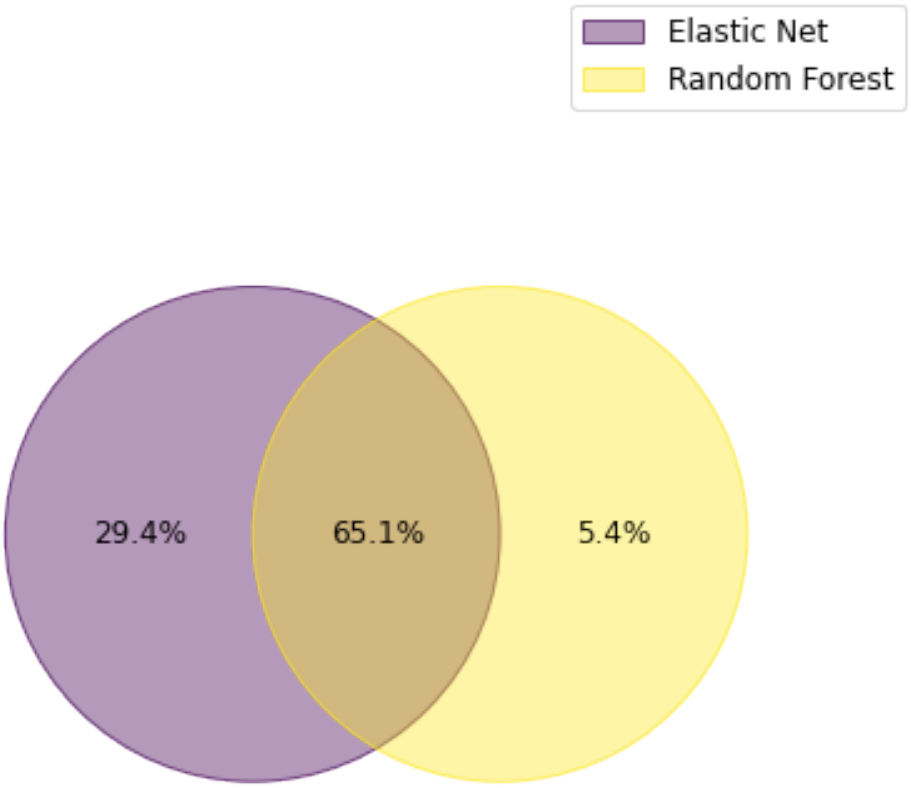
```
        txt = "There is {} out of {} and {} genes overlapping between enet and␣
↪rf - at R2 > {}"
        print(txt.format(oo, len(set(ee.feature)), len(set(rr.feature)), r2))
    print("")
```

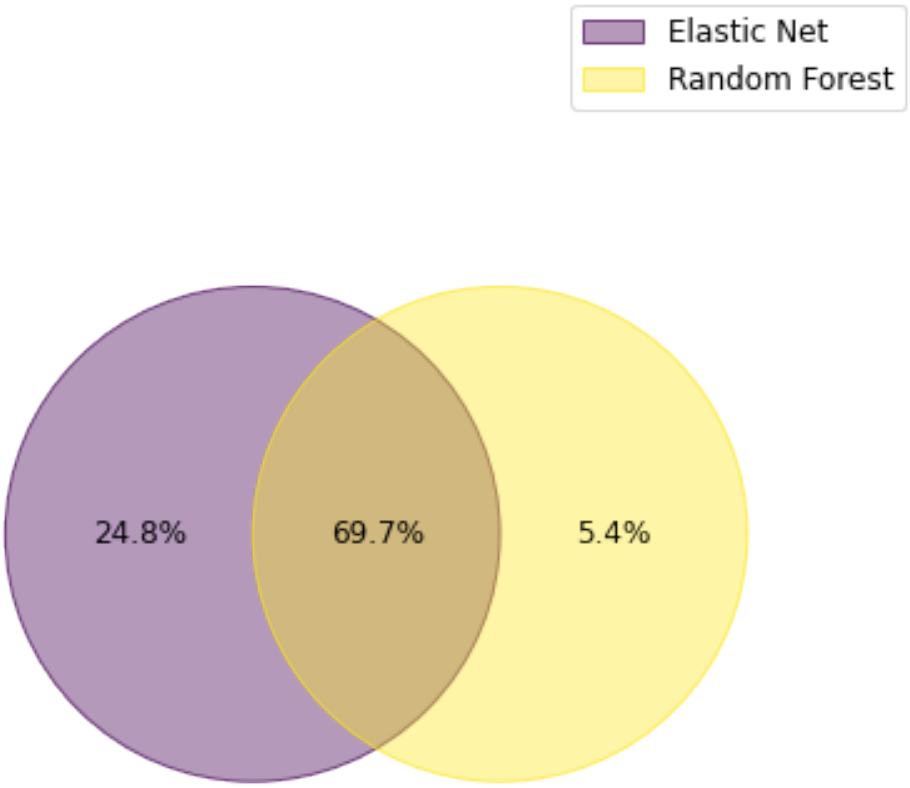
```
Caudate
There is 925 out of 1343 and 1002 genes overlapping between enet and rf - at R2
> 0
There is 320 out of 434 and 345 genes overlapping between enet and rf - at R2 >
0.2
There is 72 out of 92 and 74 genes overlapping between enet and rf - at R2 > 0.5
There is 36 out of 43 and 39 genes overlapping between enet and rf - at R2 > 0.6
There is 17 out of 18 and 19 genes overlapping between enet and rf - at R2 > 0.7
There is 9 out of 11 and 11 genes overlapping between enet and rf - at R2 > 0.75
There is 5 out of 6 and 6 genes overlapping between enet and rf - at R2 > 0.8
There is 2 out of 3 and 4 genes overlapping between enet and rf - at R2 > 0.825

DLPFC
There is 856 out of 1216 and 936 genes overlapping between enet and rf - at R2 >
0
There is 311 out of 414 and 330 genes overlapping between enet and rf - at R2 >
0.2
There is 63 out of 92 and 69 genes overlapping between enet and rf - at R2 > 0.5
There is 28 out of 41 and 30 genes overlapping between enet and rf - at R2 > 0.6
There is 13 out of 19 and 13 genes overlapping between enet and rf - at R2 > 0.7
There is 5 out of 11 and 7 genes overlapping between enet and rf - at R2 > 0.75
There is 1 out of 1 and 3 genes overlapping between enet and rf - at R2 > 0.8
There is 1 out of 1 and 2 genes overlapping between enet and rf - at R2 > 0.825

Hippocampus
There is 767 out of 1203 and 841 genes overlapping between enet and rf - at R2 >
0
There is 243 out of 335 and 263 genes overlapping between enet and rf - at R2 >
0.2
There is 46 out of 56 and 52 genes overlapping between enet and rf - at R2 > 0.5
There is 26 out of 32 and 29 genes overlapping between enet and rf - at R2 > 0.6
There is 13 out of 15 and 15 genes overlapping between enet and rf - at R2 > 0.7
There is 12 out of 12 and 13 genes overlapping between enet and rf - at R2 >
0.75
There is 5 out of 7 and 6 genes overlapping between enet and rf - at R2 > 0.8
There is 3 out of 4 and 4 genes overlapping between enet and rf - at R2 > 0.825

Dentate Gyrus
There is 175 out of 263 and 216 genes overlapping between enet and rf - at R2 >
0
There is 72 out of 90 and 94 genes overlapping between enet and rf - at R2 > 0.2
There is 12 out of 20 and 14 genes overlapping between enet and rf - at R2 > 0.5
There is 5 out of 6 and 7 genes overlapping between enet and rf - at R2 > 0.6
```

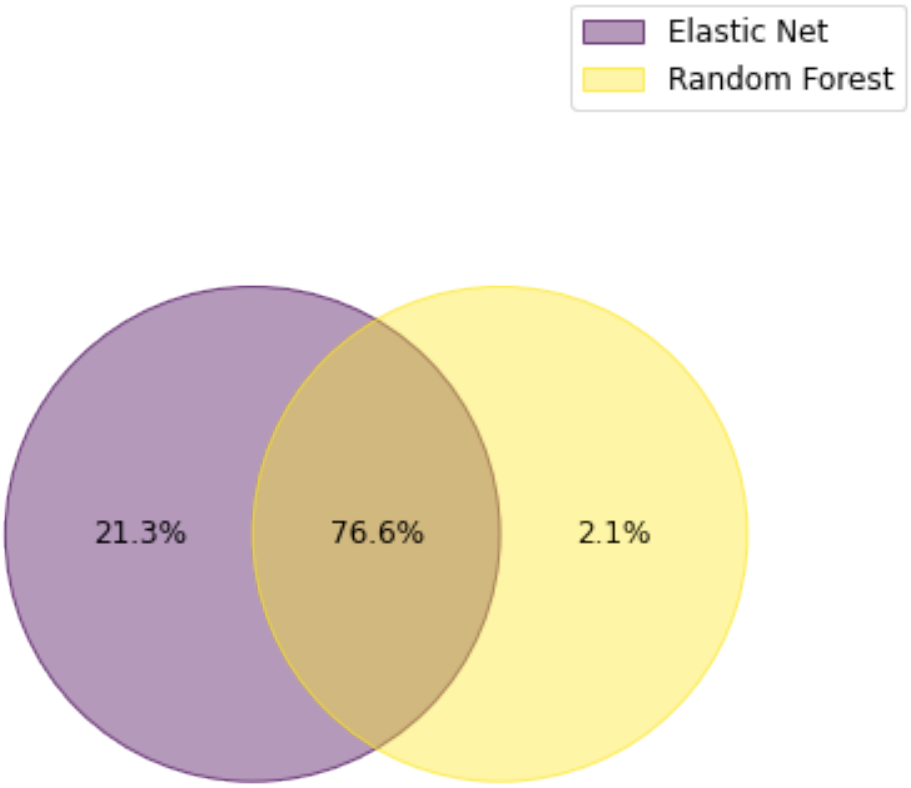
```
There is 1 out of 3 and 1 genes overlapping between enet and rf - at R2 > 0.7
There is 1 out of 1 and 1 genes overlapping between enet and rf - at R2 > 0.75
There is 0 out of 0 and 1 genes overlapping between enet and rf - at R2 > 0.8
There is 0 out of 0 and 1 genes overlapping between enet and rf - at R2 > 0.825
```

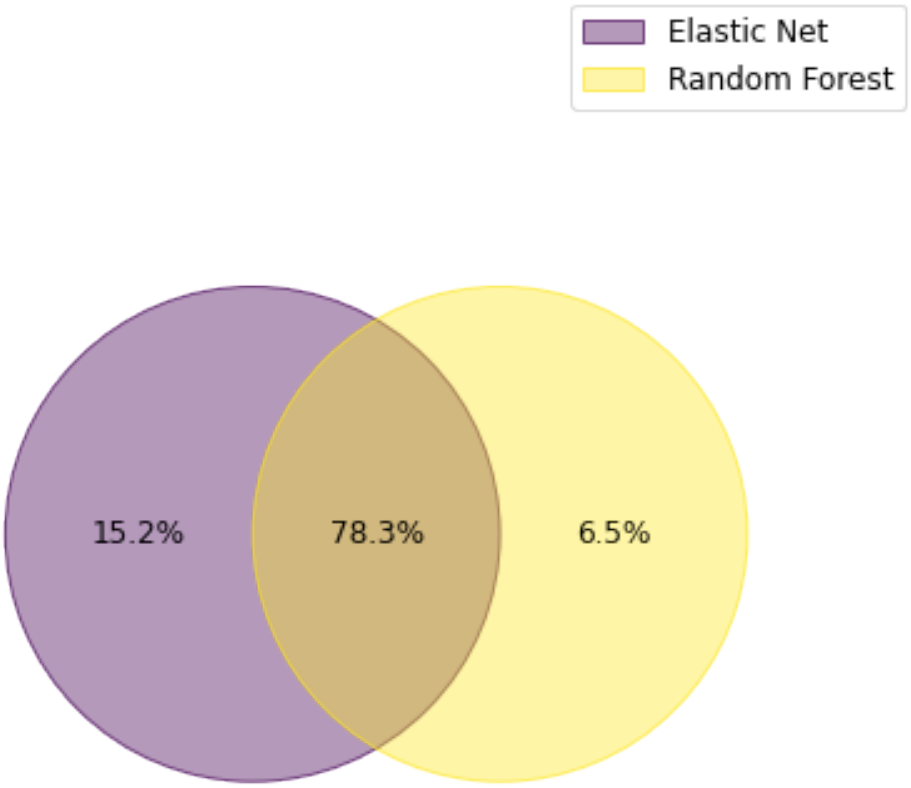
[16]:
```
dirname = "model_venn_diagrams"
mkdir_p(dirname)
for tissue in ["Caudate", "DLPFC", "Hippocampus", "Dentate Gyrus"]:
    #print(tissue)
    for r2 in [0, 0.2, 0.5, 0.6, 0.7, 0.75, 0.8]:
        ee = enet[(enet["tissue"] == tissue) & (enet["test_score_r2"] >= r2)].
↪copy()
        rr = rf[(rf["tissue"] == tissue) & (rf["test_score_r2"] >= r2)].copy()
        model_set = {"Elastic Net": set(ee.feature), "Random Forest": set(rr.
↪feature),}
        venn(model_set, fmt="{percentage:.1f}%", fontsize=12)
        tt = tissue.lower().replace(" ", "_")
        plt.savefig("{}/venn_diagram_modelOverlap_{}_r2_{}.png".format(dirname,␣
↪tt, r2))
        plt.savefig("{}/venn_diagram_modelOverlap_{}_r2_{}.pdf".format(dirname,␣
↪tt, r2))
        plt.savefig("{}/venn_diagram_modelOverlap_{}_r2_{}.svg".format(dirname,␣
↪tt, r2))
```

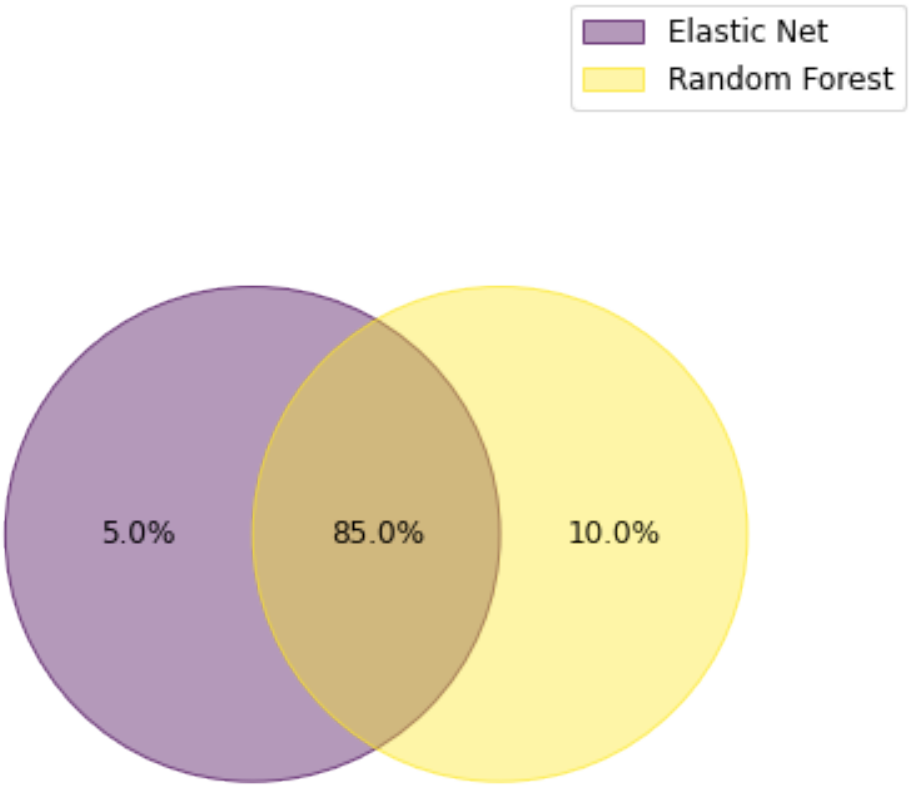
```
/home/jbenja13/.local/lib/python3.9/site-packages/venn/_venn.py:83:
RuntimeWarning: More than 20 figures have been opened. Figures created through
the pyplot interface (`matplotlib.pyplot.figure`) are retained until explicitly
closed and may consume too much memory. (To control this warning, see the
rcParam `figure.max_open_warning`).
  _, ax = subplots(nrows=1, ncols=1, figsize=figsize)
```

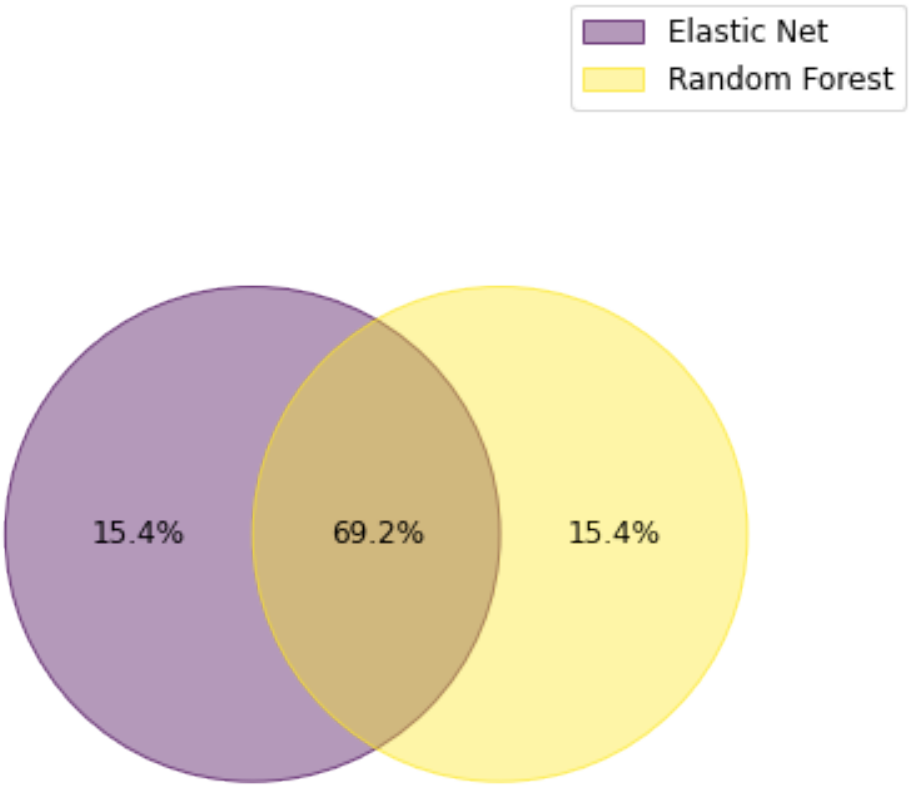

Elastic Net
Random Forest
29.4%
65.1%
5.4%

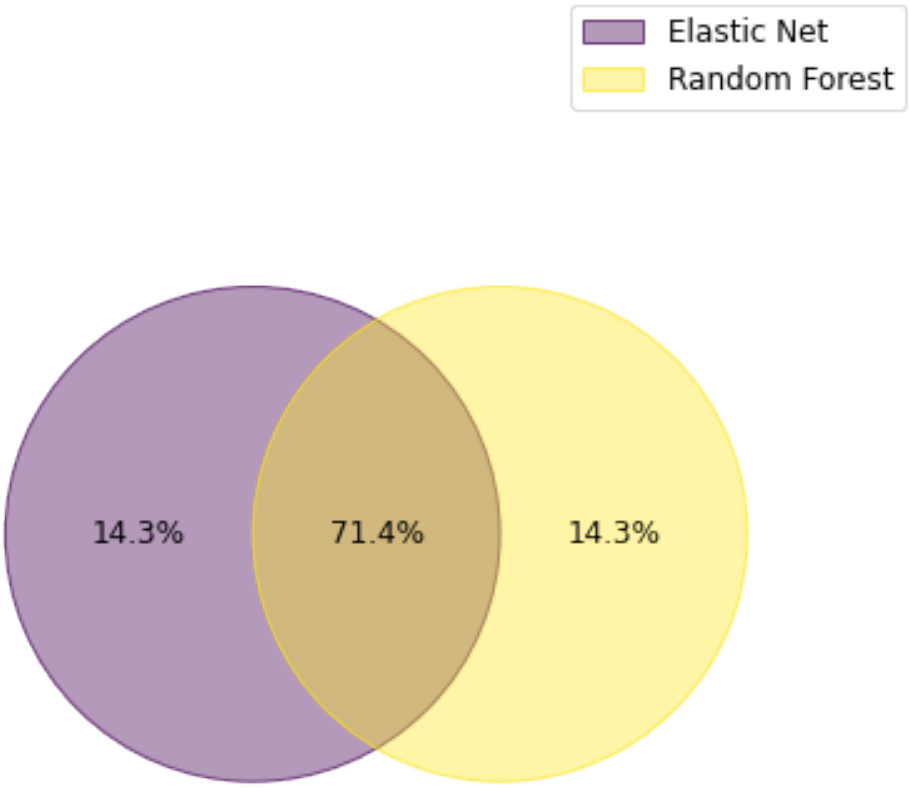

Elastic Net
Random Forest
24.8%
69.7%
5.4%

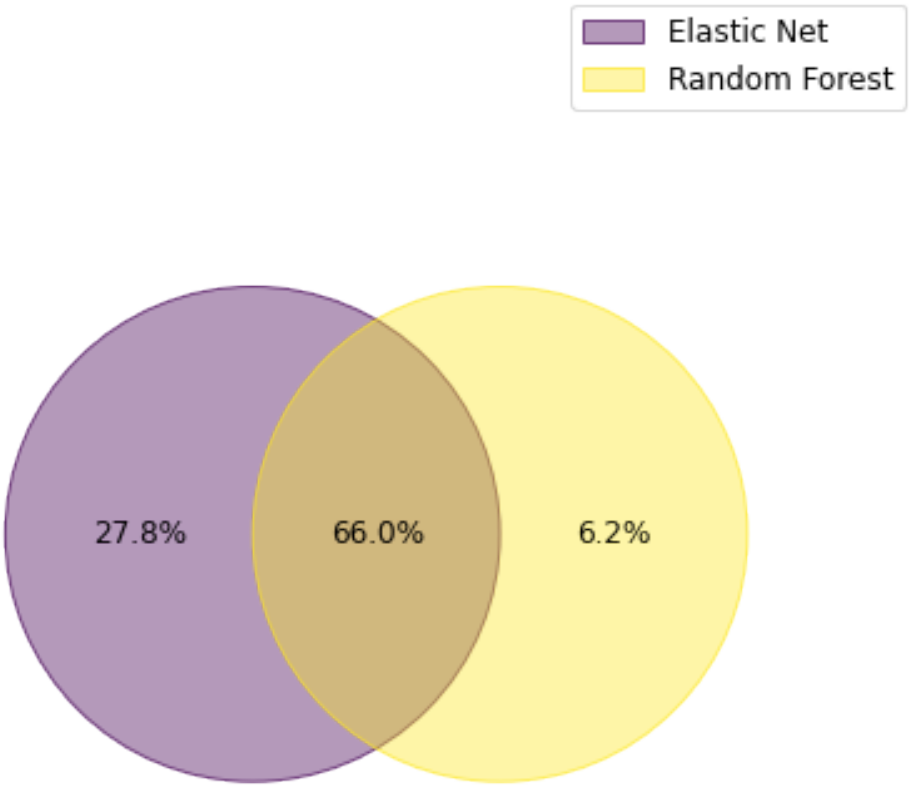

Elastic Net
Random Forest
21.3%
76.6%
2.1%

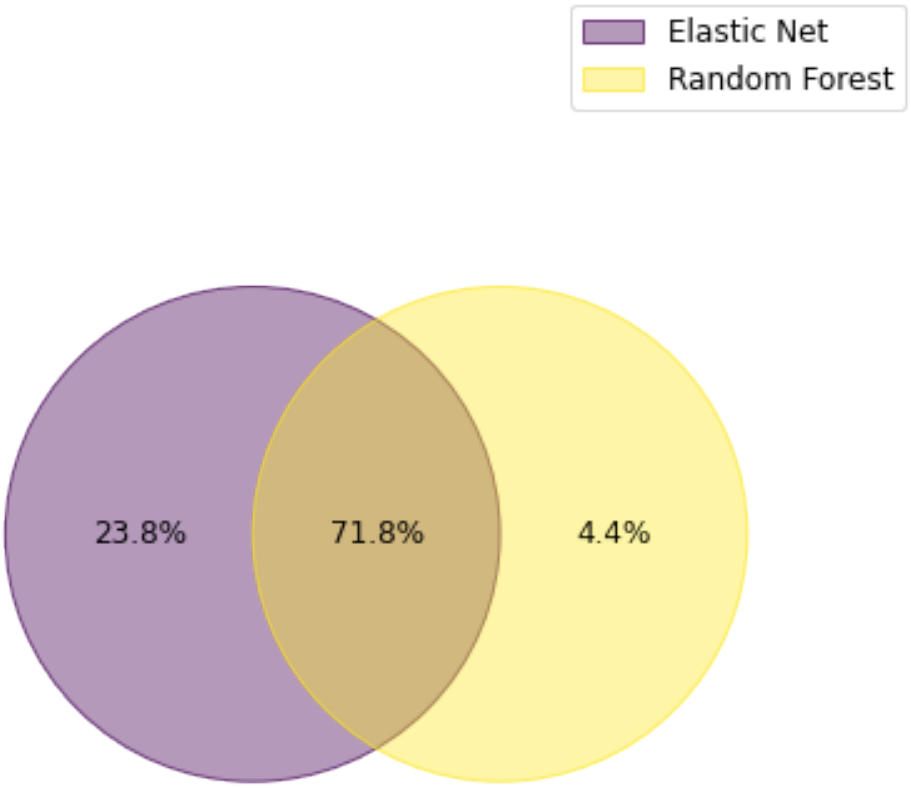

Elastic Net
Random Forest
15.2%
78.3%
6.5%

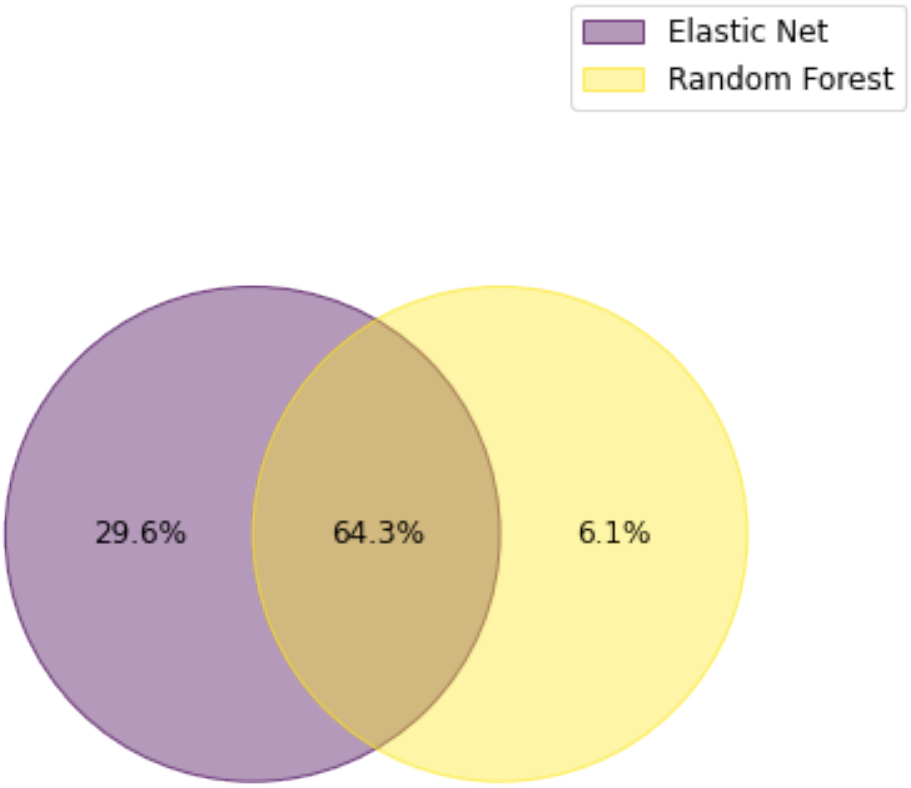

Elastic Net
Random Forest
5.0%
85.0%
10.0%

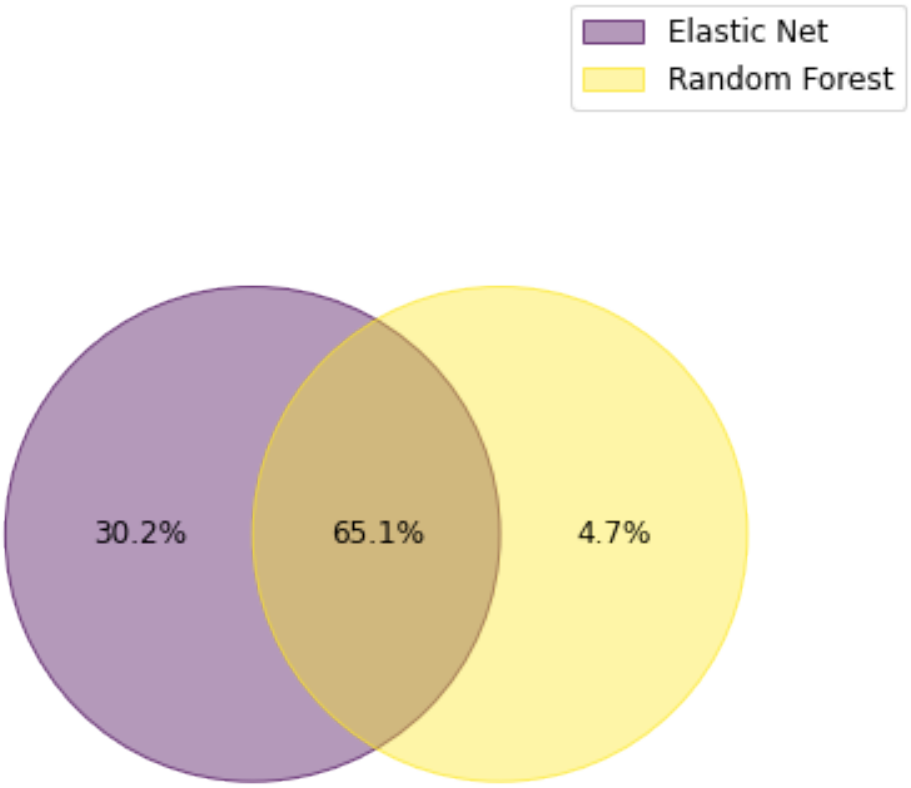

Elastic Net
Random Forest
15.4%
69.2%
15.4%

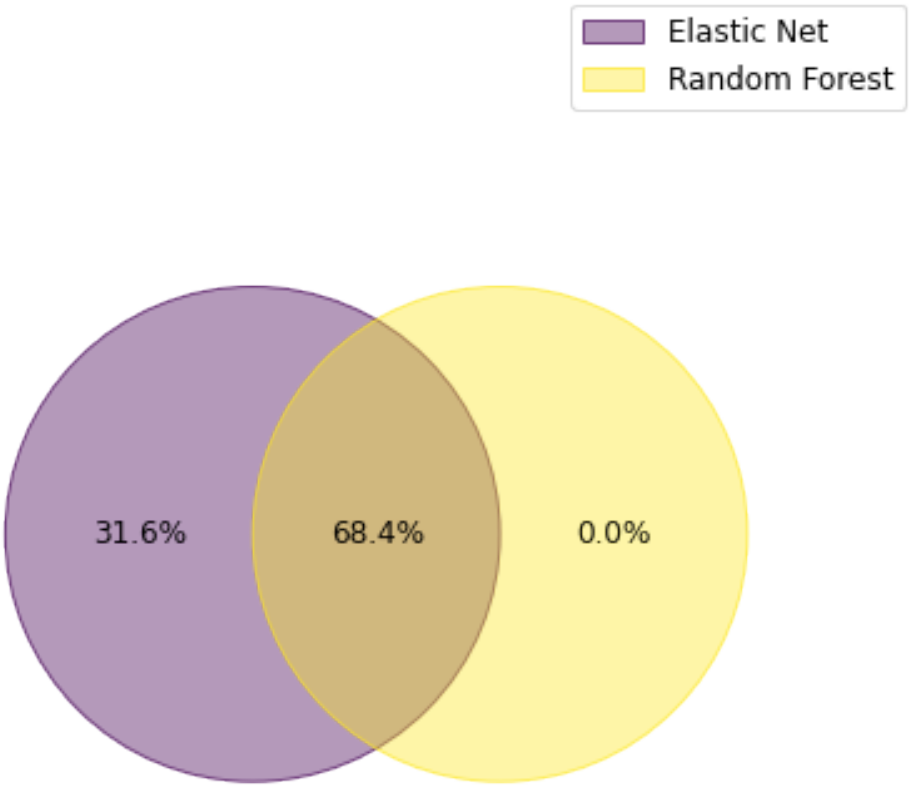

Elastic Net
Random Forest
14.3%
71.4%
14.3%

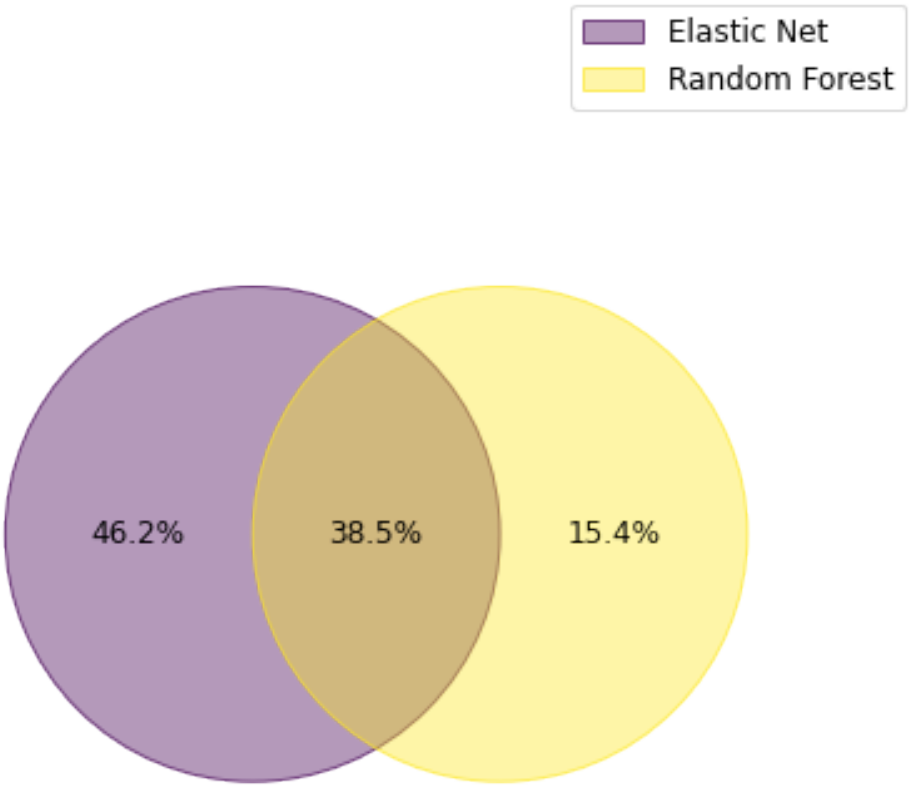

Elastic Net
Random Forest
27.8%
66.0%
6.2%

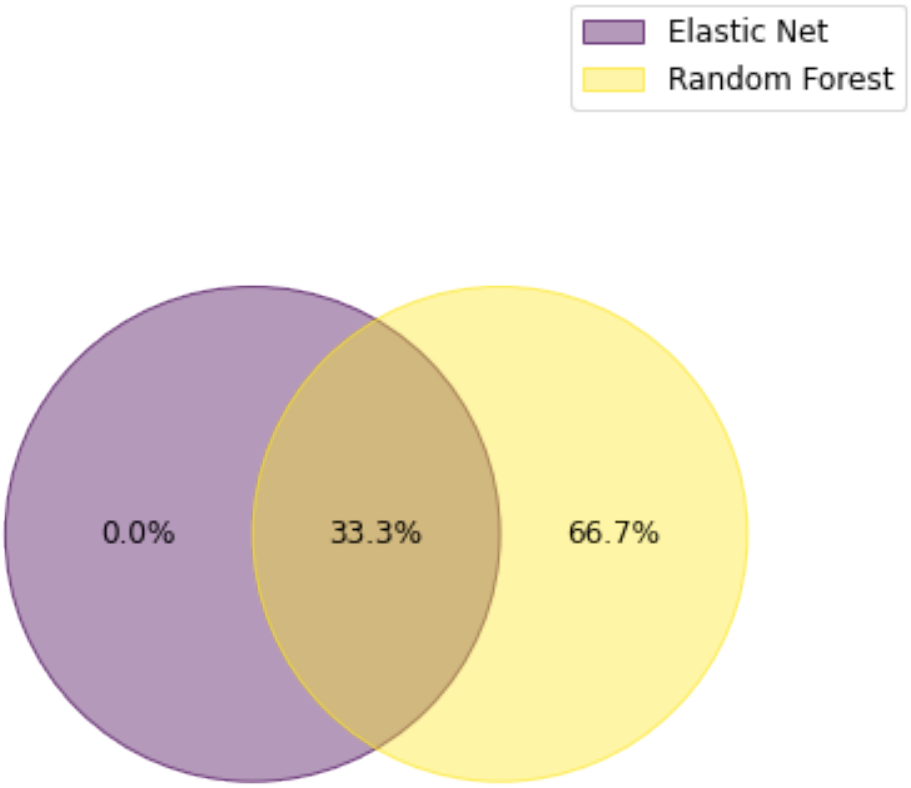

Elastic Net
Random Forest
23.8%
71.8%
4.4%

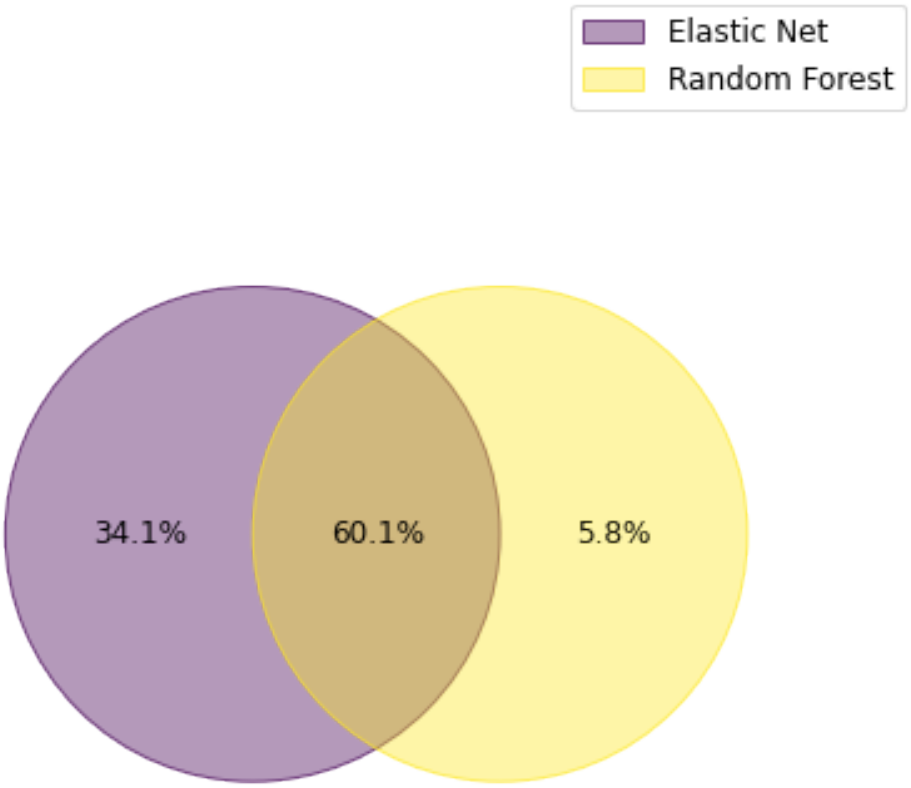

Elastic Net
Random Forest
29.6%
64.3%
6.1%

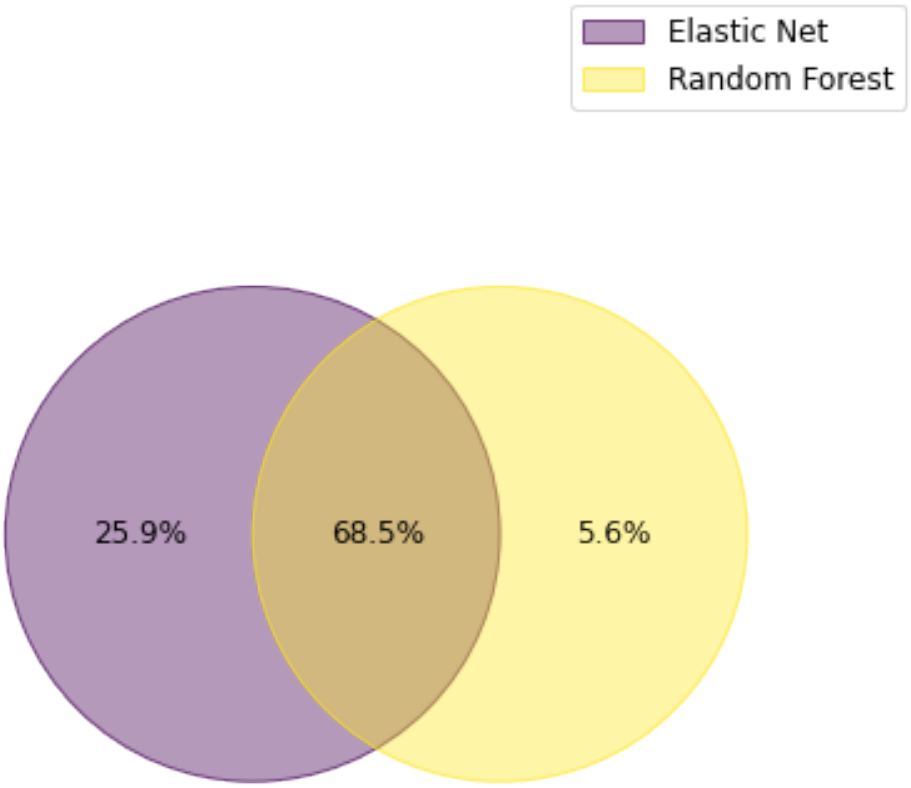

Elastic Net
Random Forest
30.2%
65.1%
4.7%

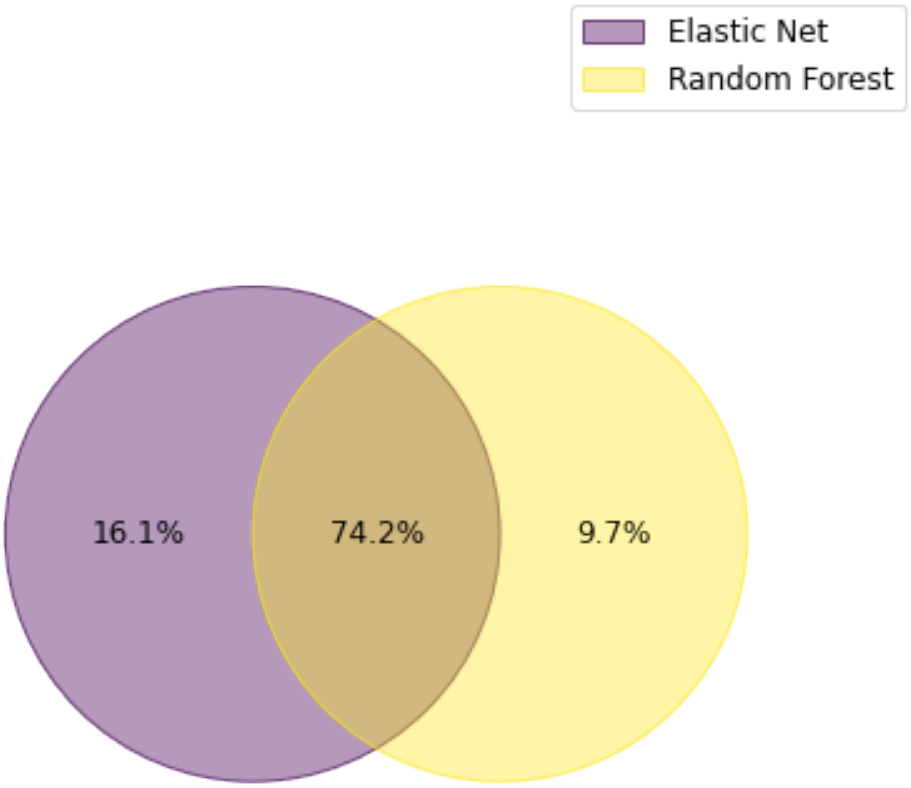

Elastic Net
Random Forest
31.6%
68.4%
0.0%

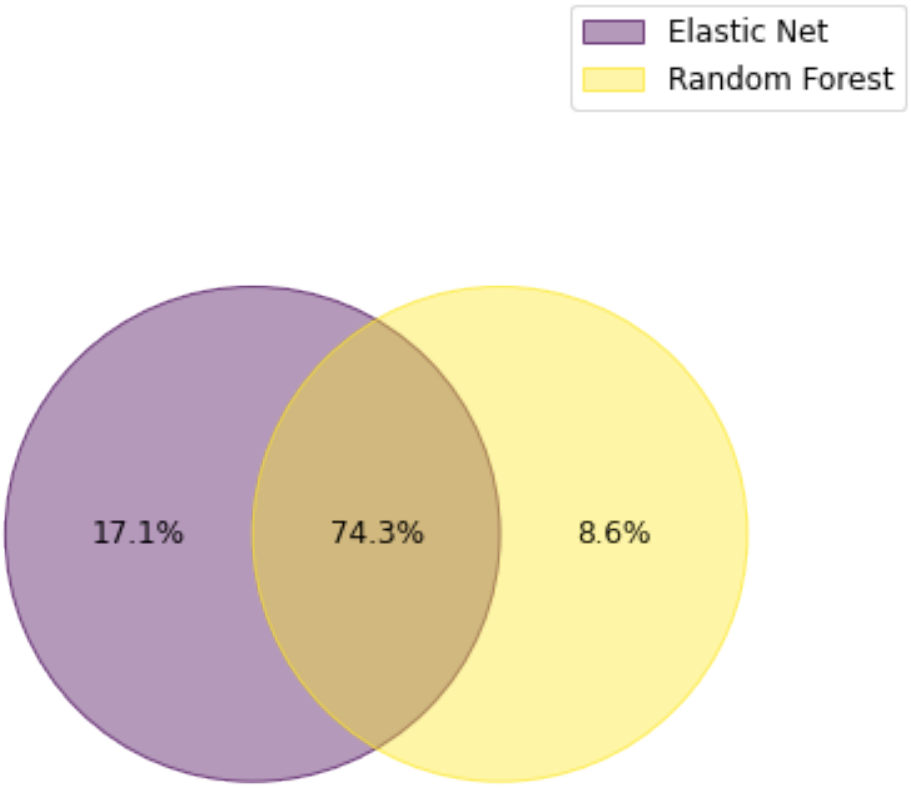

Elastic Net
Random Forest
46.2%
38.5%
15.4%

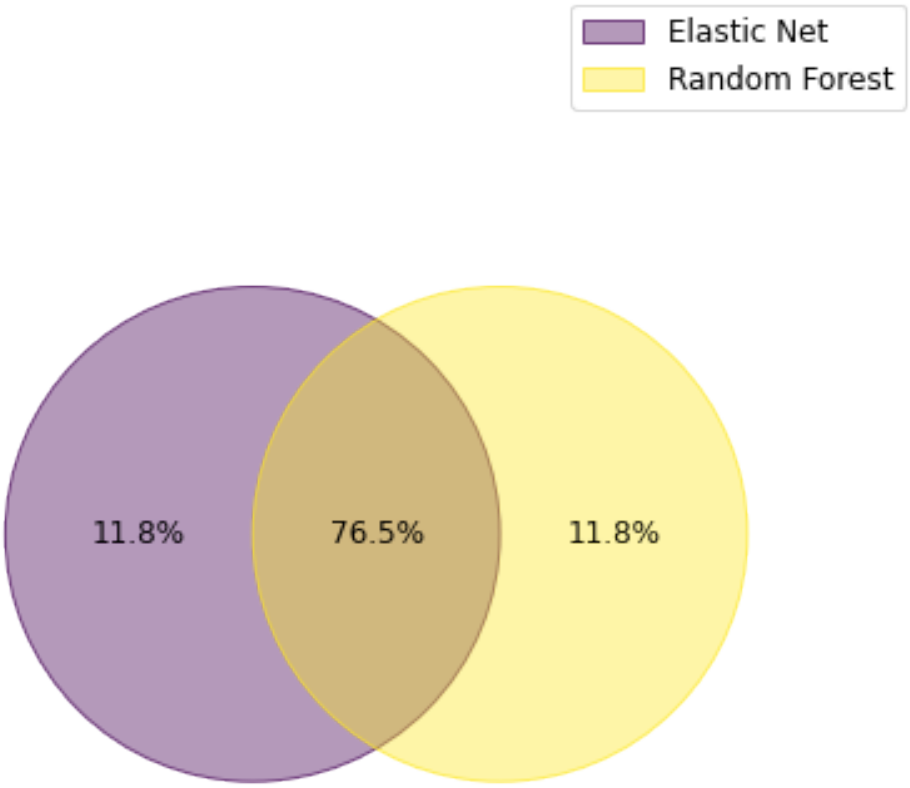

Elastic Net
Random Forest
0.0%
33.3%
66.7%

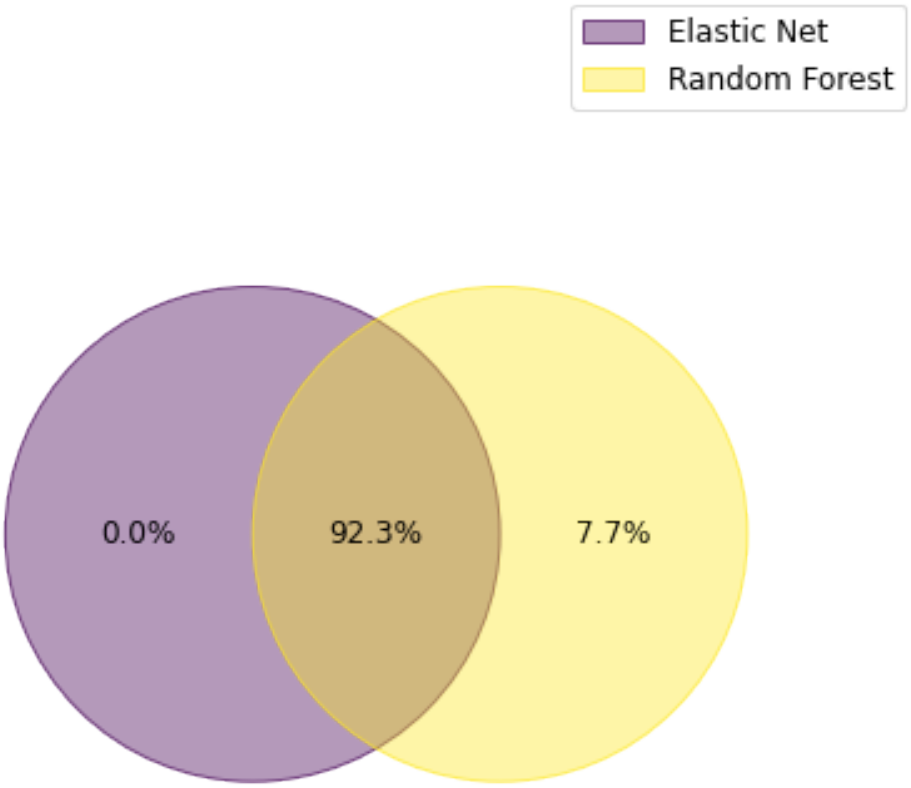

Elastic Net
Random Forest
34.1%
60.1%
5.8%

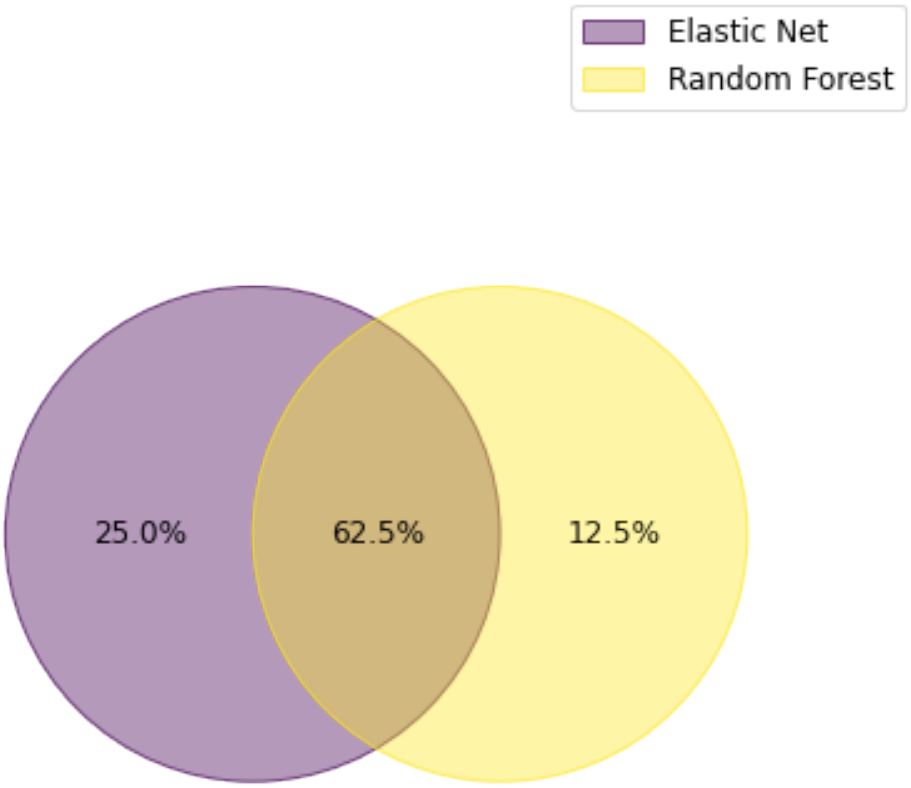

Elastic Net
Random Forest
25.9%
68.5%
5.6%

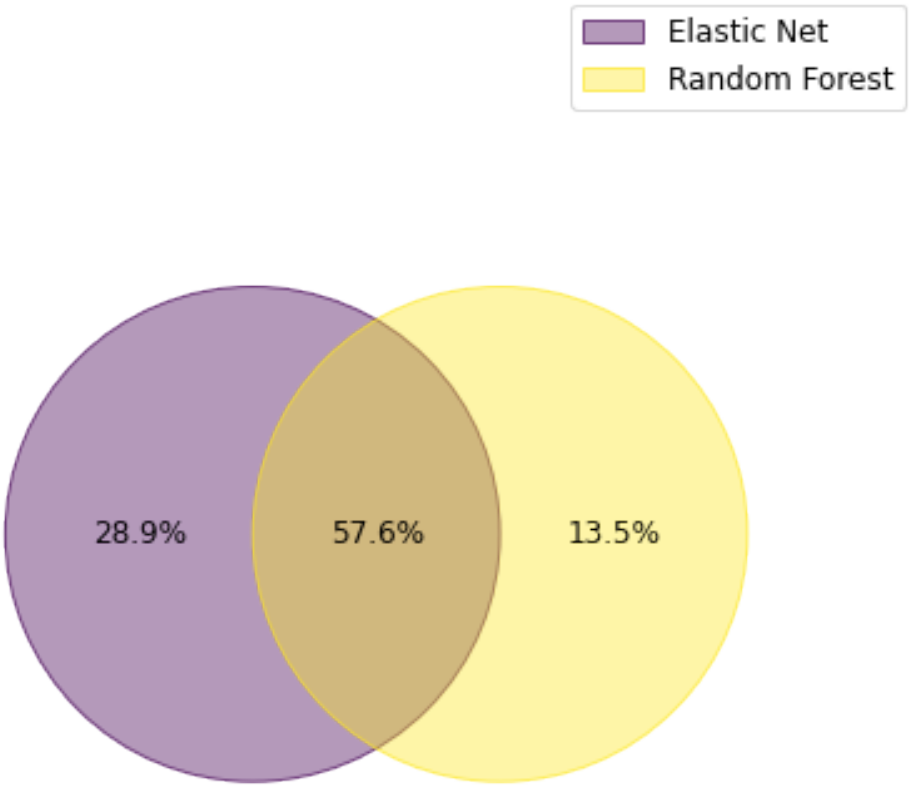

Elastic Net
Random Forest
16.1%
74.2%
9.7%

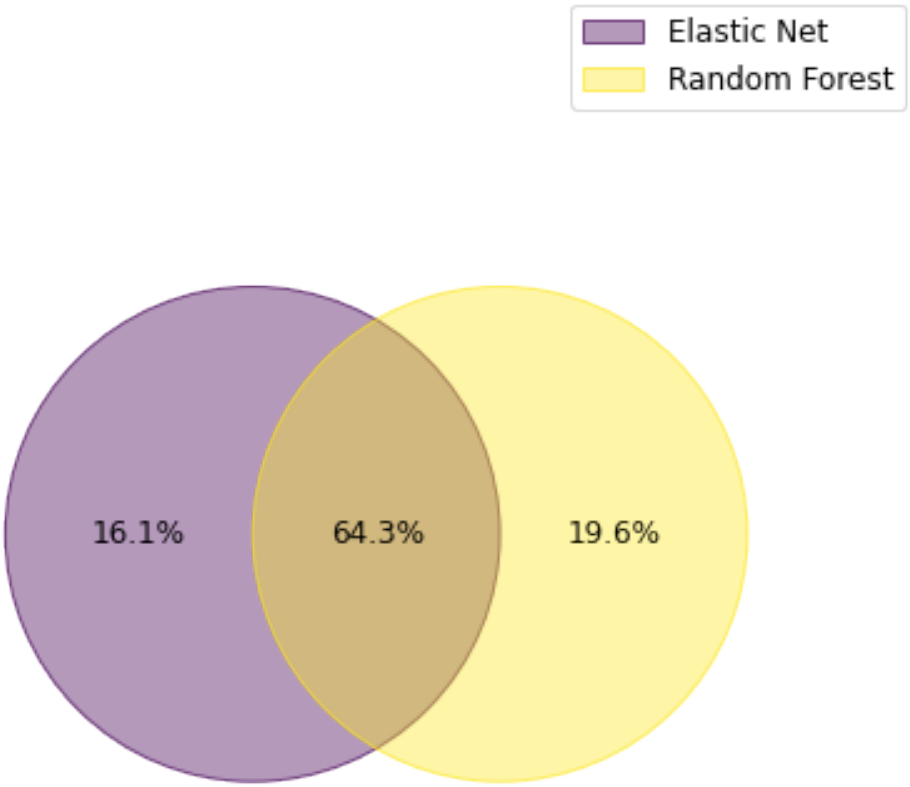

Elastic Net
Random Forest
17.1%
74.3%
8.6%

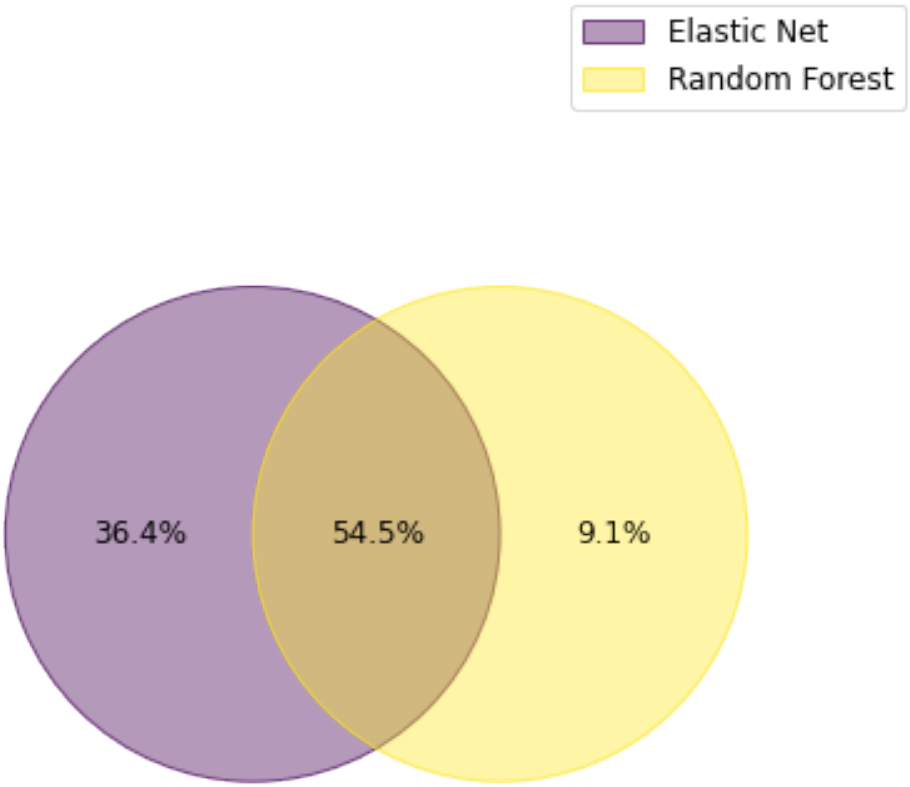

Elastic Net
Random Forest
11.8%
76.5%
11.8%

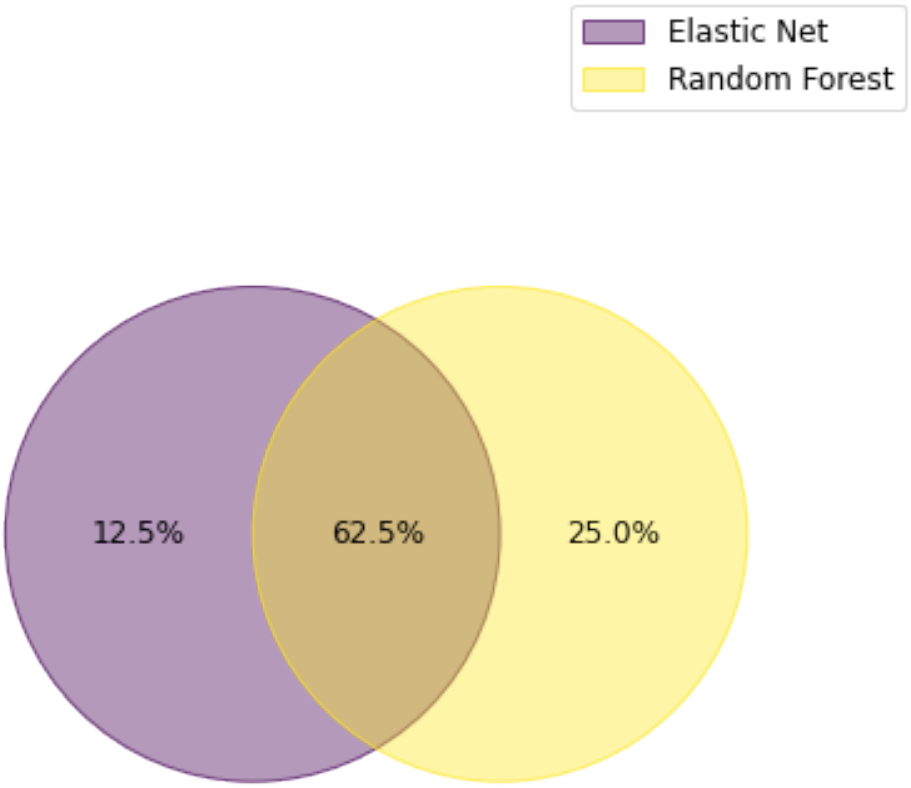

Elastic Net
Random Forest
0.0%
92.3%
7.7%

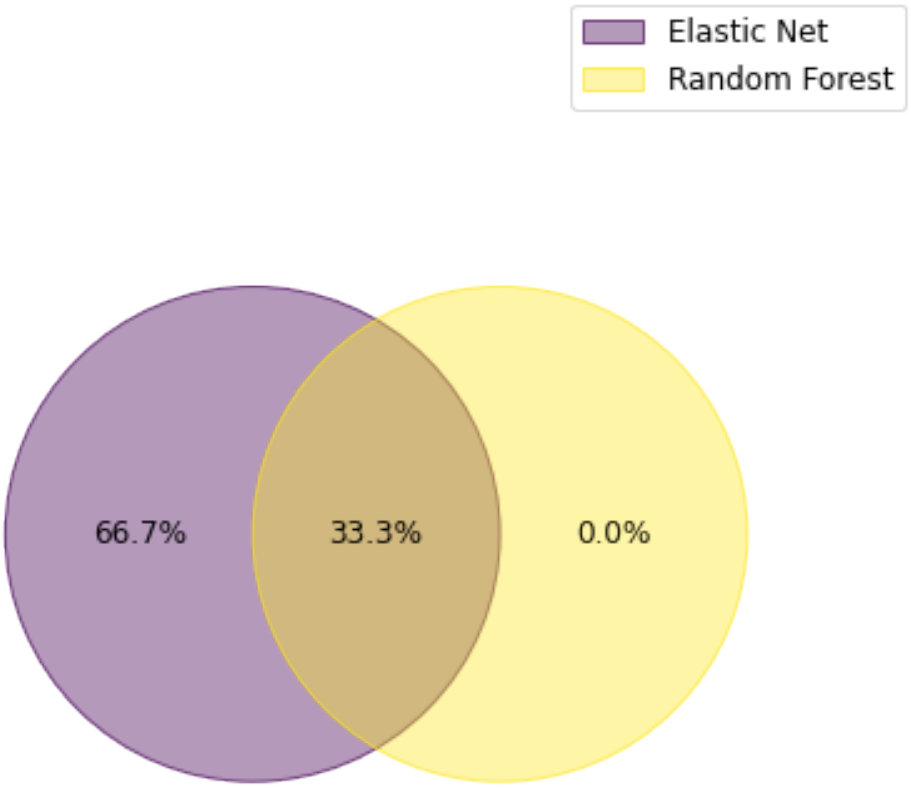

Elastic Net
Random Forest
25.0%
62.5%
12.5%

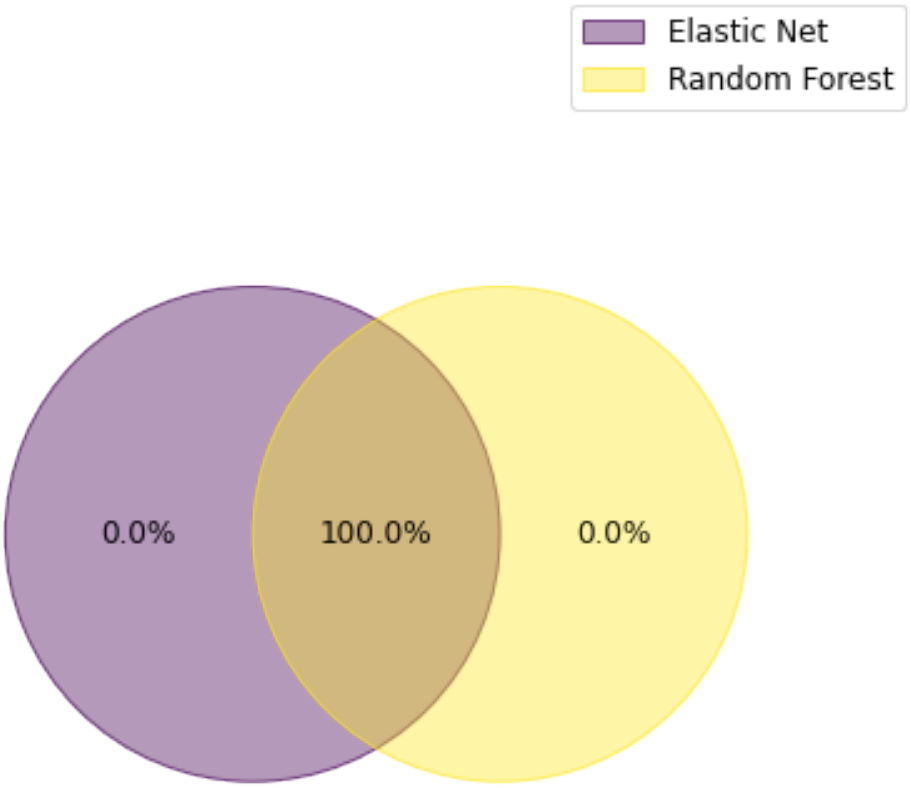

Elastic Net
Random Forest
28.9%
57.6%
13.5%

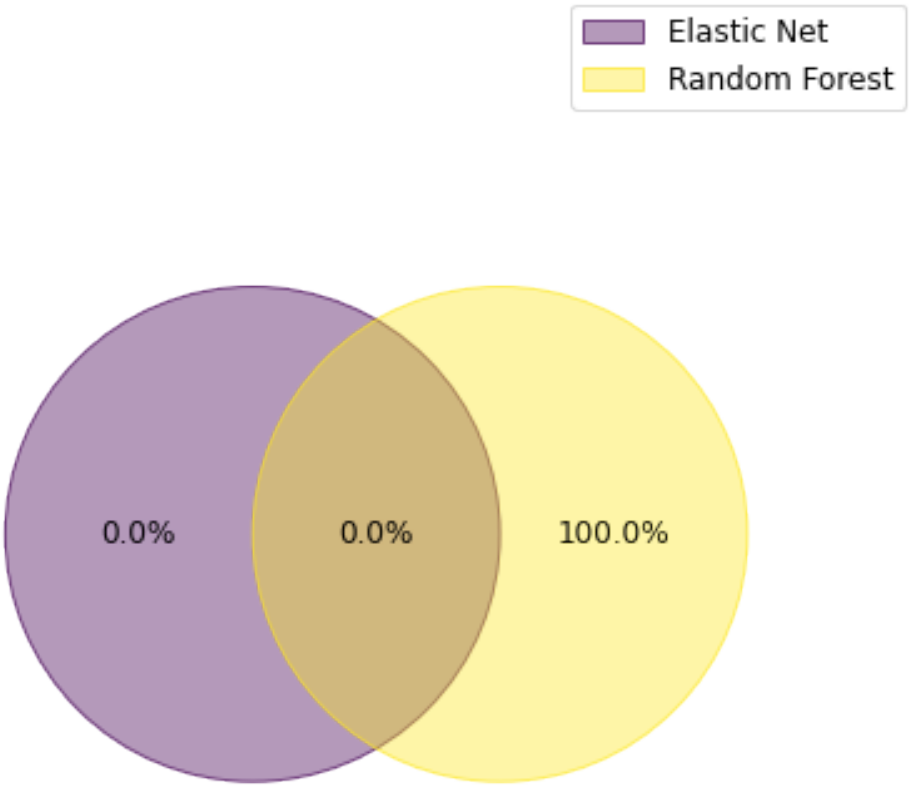

Elastic Net
Random Forest
16.1%
64.3%
19.6%


Elastic Net
Random Forest
36.4%
54.5%
9.1%


Elastic Net
Random Forest
12.5%
62.5%
25.0%


Elastic Net
Random Forest
66.7%
33.3%
0.0%

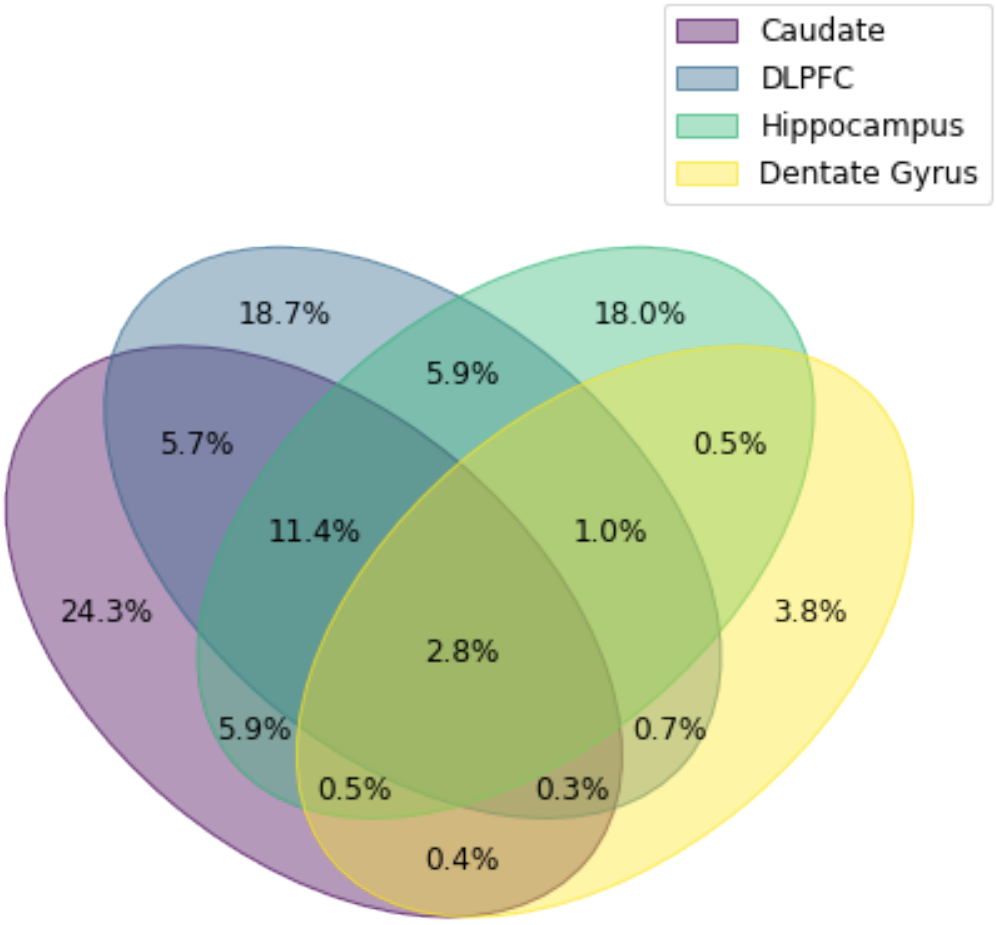

Elastic Net
Random Forest
0.0%
100.0%
0.0%

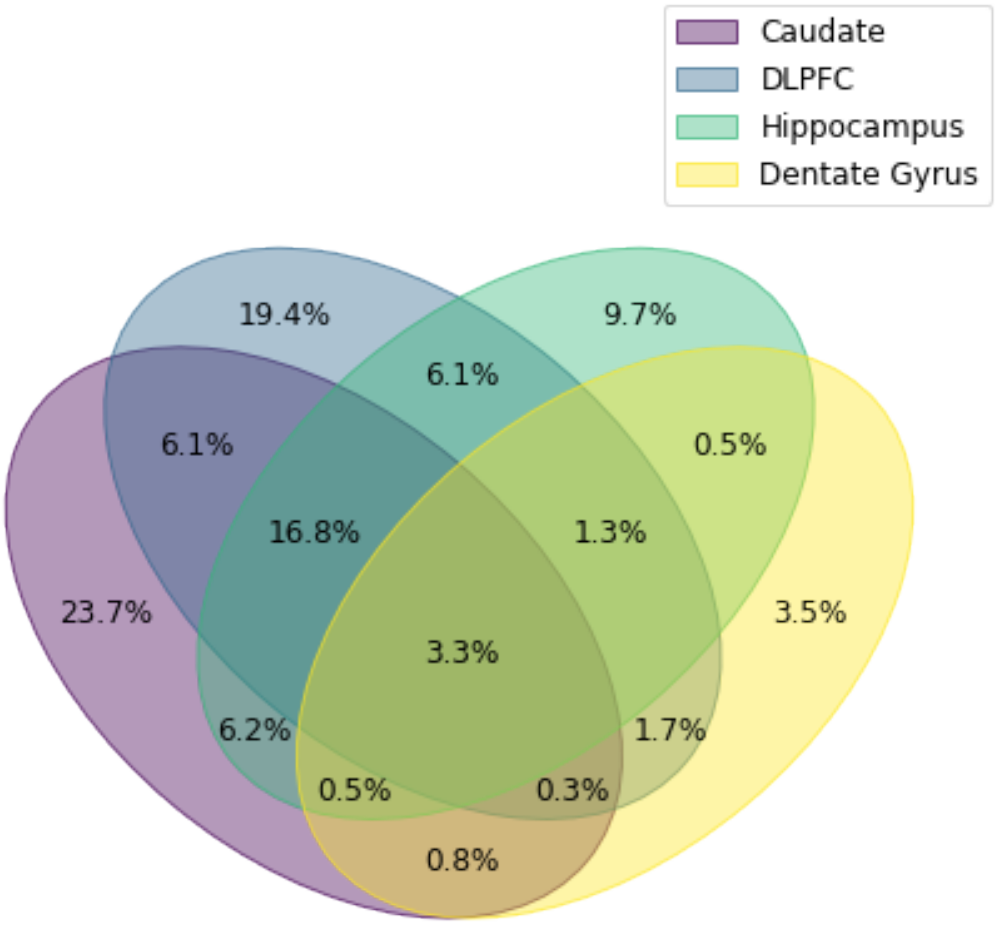


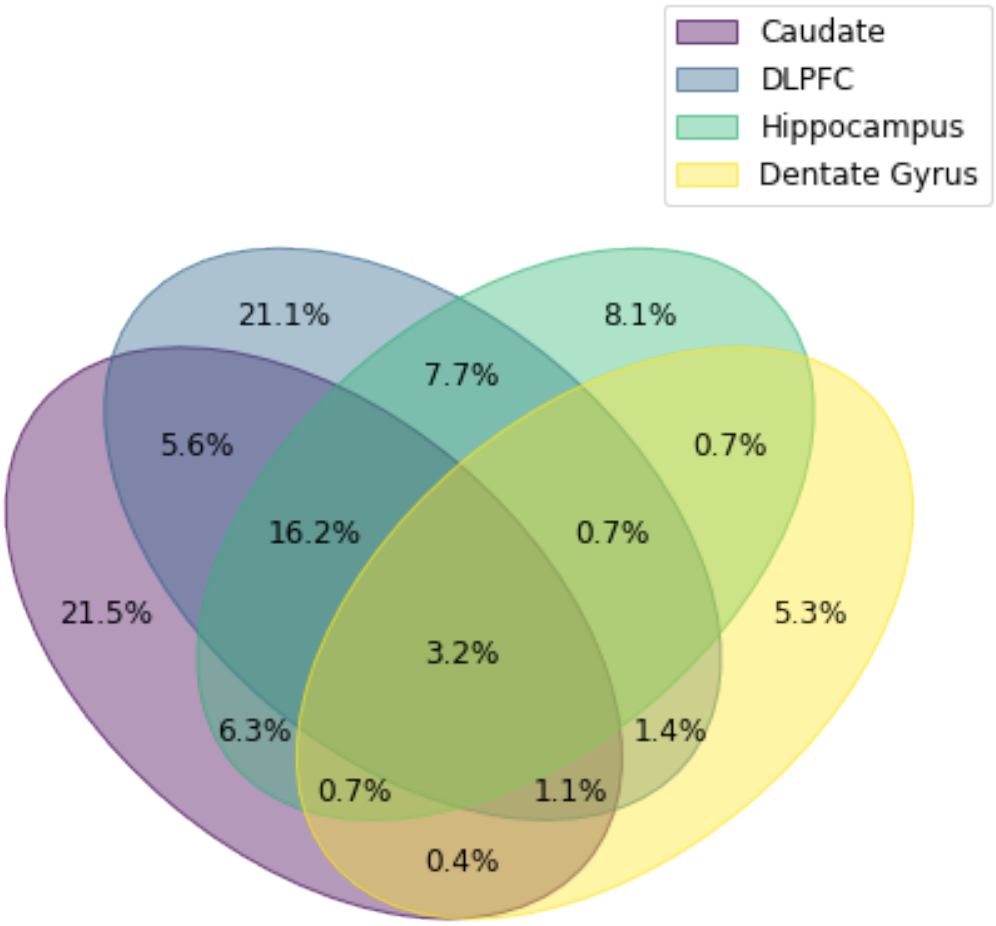
1.3.5 What is the overlap between brain regions?

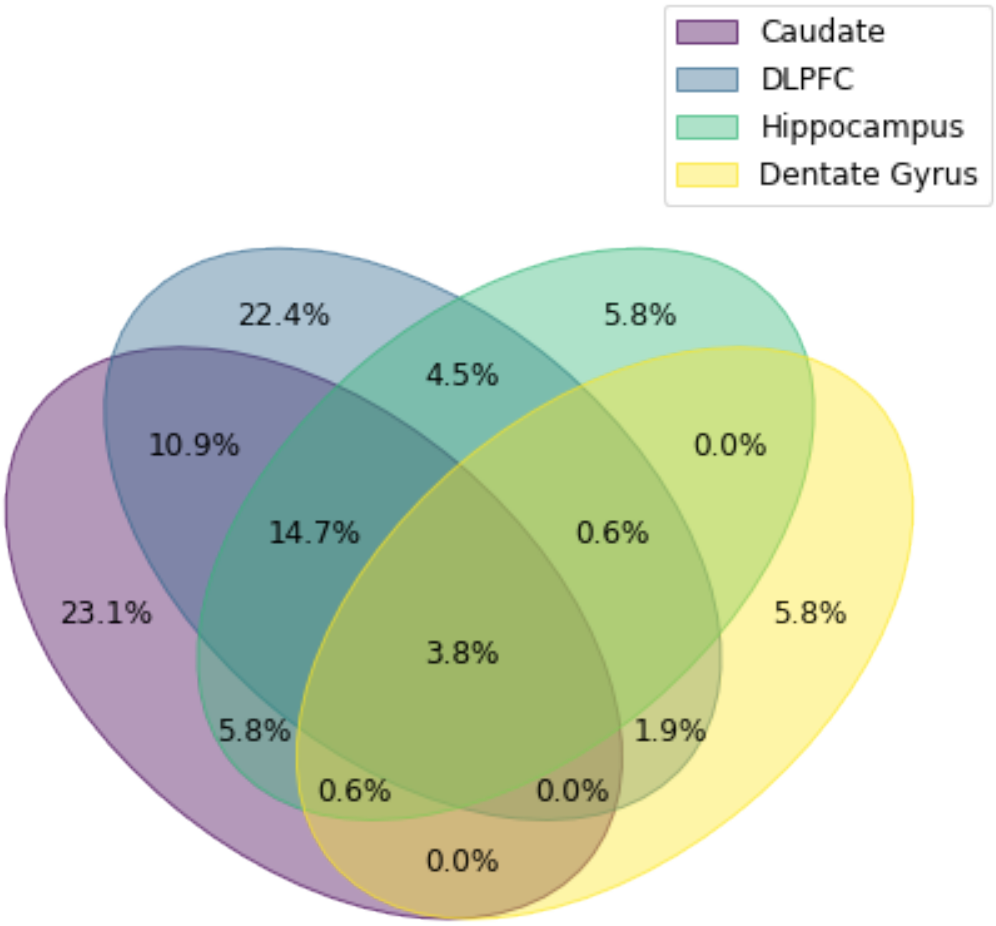
[17]:
```
dirname = "tissue_venn_diagrams"
mkdir_p(dirname)
for modeln in ["Elastic Net", "Random Forest"]:
    #print(modeln)
    dft = df[(df['Model'] == modeln)].copy()
    for r2 in [0, 0.2, 0.4, 0.5, 0.6, 0.7, 0.75, 0.8]:
        cc = dft[(dft["tissue"] == "Caudate") & (dft["test_score_r2"] >= r2)].
↪copy()
        dd = dft[(dft["tissue"] == "DLPFC") & (dft["test_score_r2"] >= r2)].
↪copy()
```

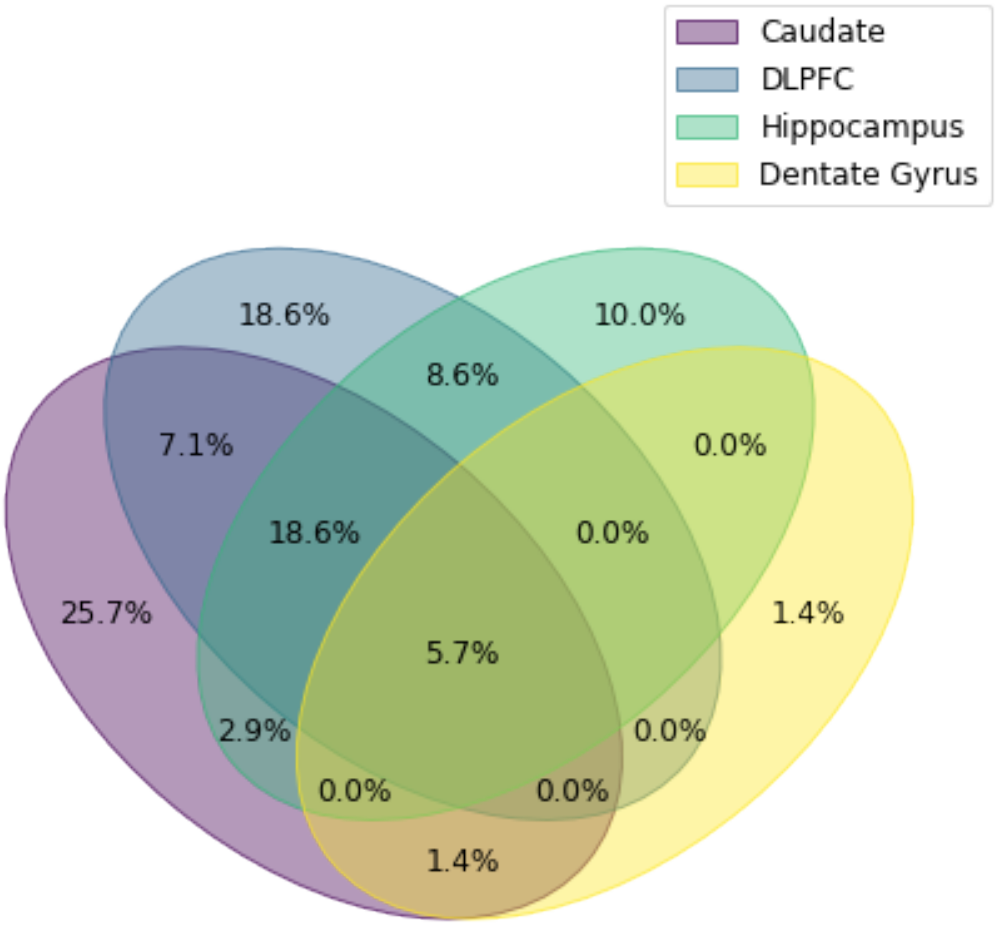
```
        hh = dft[(dft["tissue"] == "Hippocampus") & (dft["test_score_r2"] >=␣
↪r2)].copy()
        gg = dft[(dft["tissue"] == "Dentate Gyrus") & (dft["test_score_r2"] >=␣
↪r2)].copy()
        tissues = {"Caudate": set(cc.feature), "DLPFC": set(dd.feature),
                   "Hippocampus": set(hh.feature), "Dentate Gyrus": set(gg.
↪feature)}
        venn(tissues, fmt="{percentage:.1f}%", fontsize=12)
        mm = modeln.lower().replace(" ", "_")
        plt.savefig("{}/venn_diagram_tissueOverlap_{}_r2_{}.png".
↪format(dirname, mm, r2))
        plt.savefig("{}/venn_diagram_tissueOverlap_{}_r2_{}.pdf".
↪format(dirname, mm, r2))
        plt.savefig("{}/venn_diagram_tissueOverlap_{}_r2_{}.svg".
↪format(dirname, mm, r2))
```

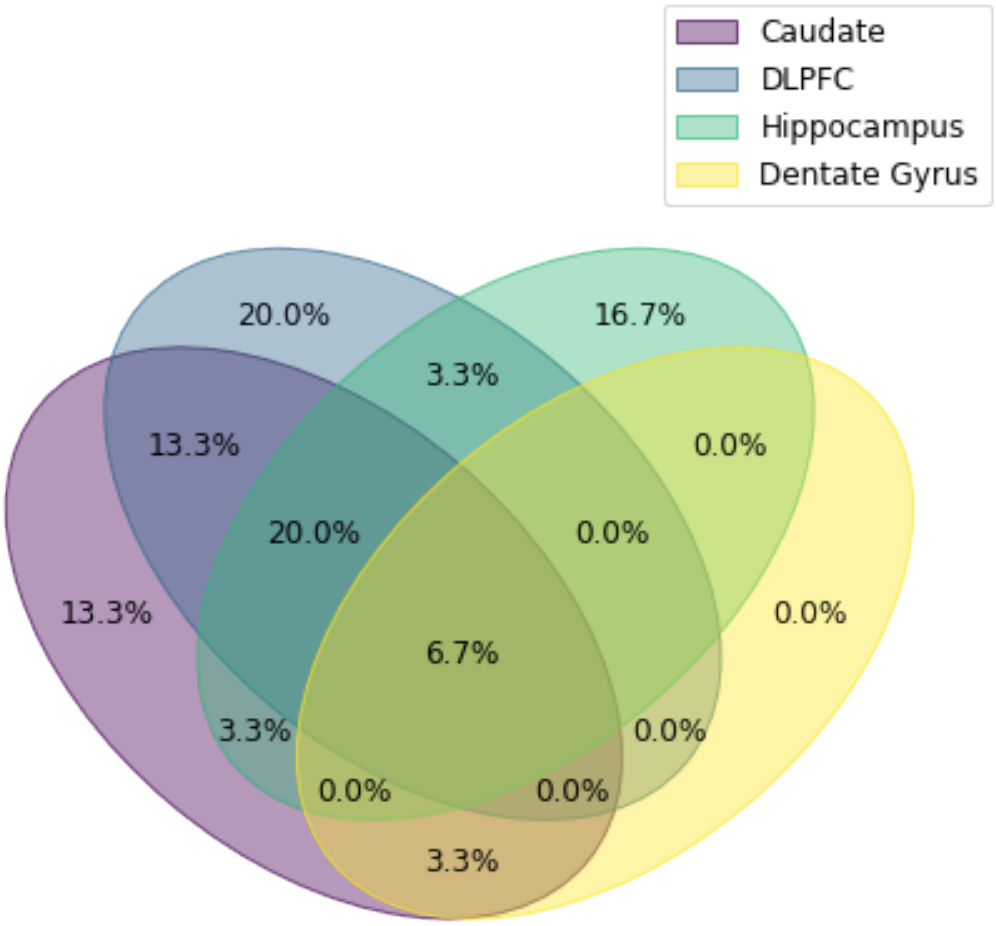


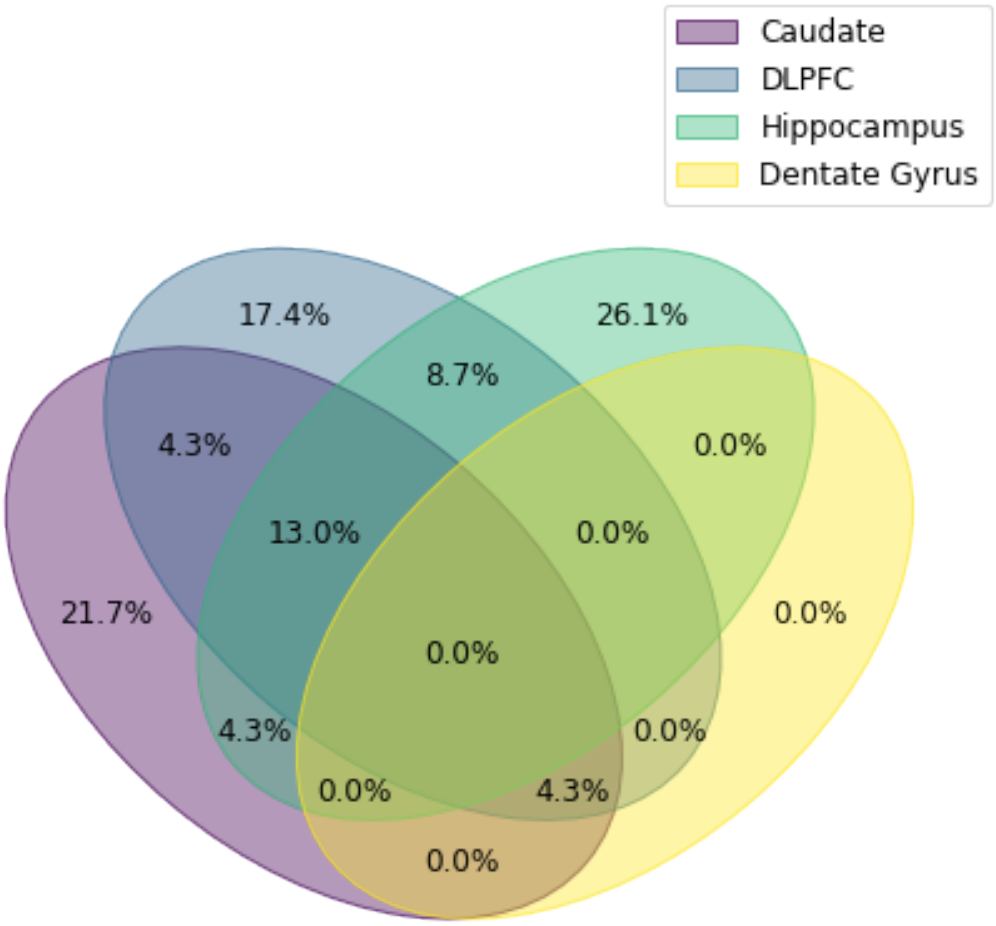

Caudate
DLPFC
Hippocampus
Dentate Gyrus
19.4%
9.7%
6.1%
6.1%
0.5%
16.8%
1.3%
23.7%
3.5%
3.3%
6.2%
1.7%
0.5%
0.3%
0.8%

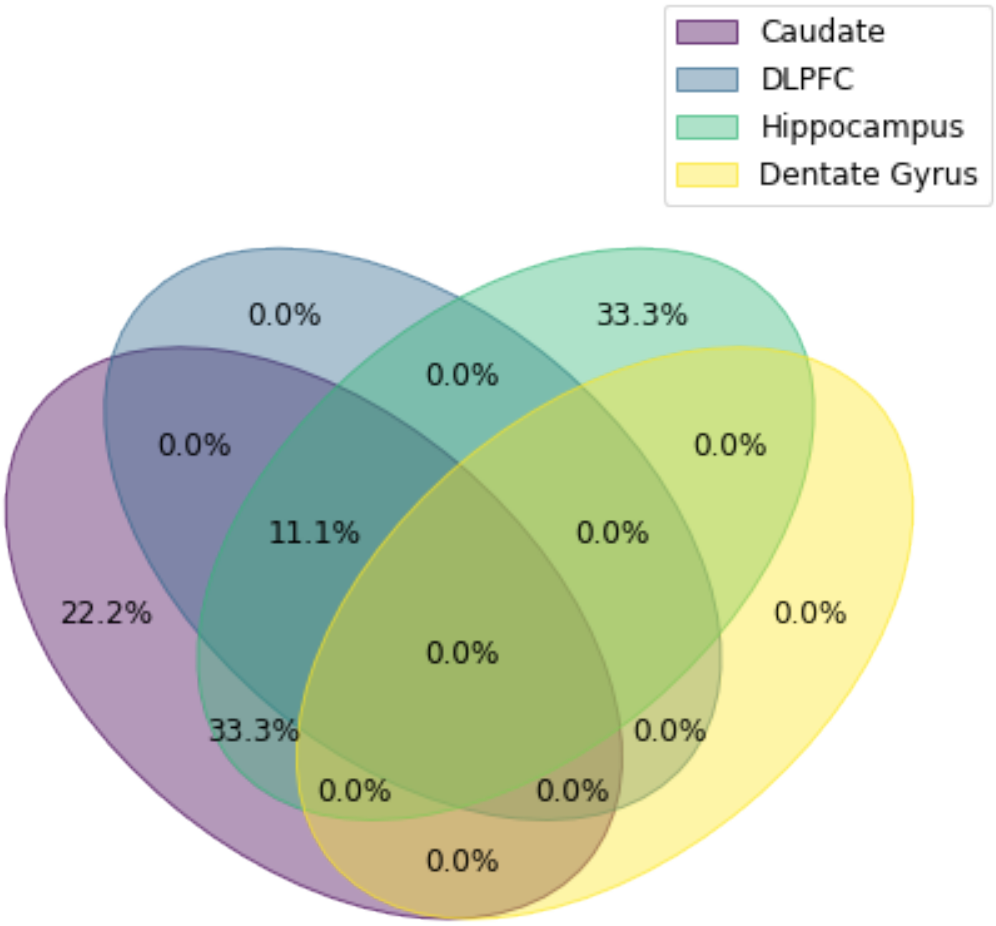

Caudate
DLPFC
Hippocampus
Dentate Gyrus
21.1%
8.1%
7.7%
5.6%
0.7%
16.2%
0.7%
21.5%
5.3%
3.2%
6.3%
1.4%
0.7%
1.1%
0.4%

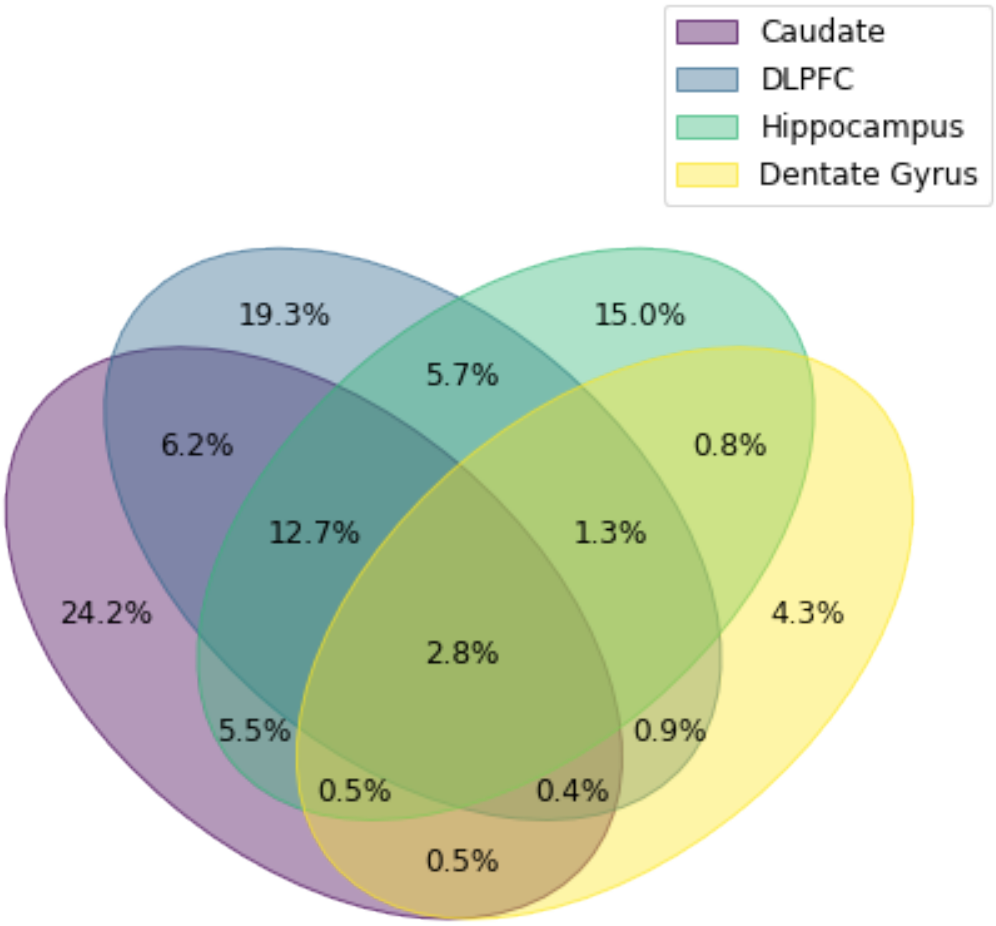

Caudate
DLPFC
Hippocampus
Dentate Gyrus
22.4%
5.8%
4.5%
10.9%
0.0%
14.7%
0.6%
23.1%
5.8%
3.8%
5.8%
1.9%
0.6%
0.0%
0.0%

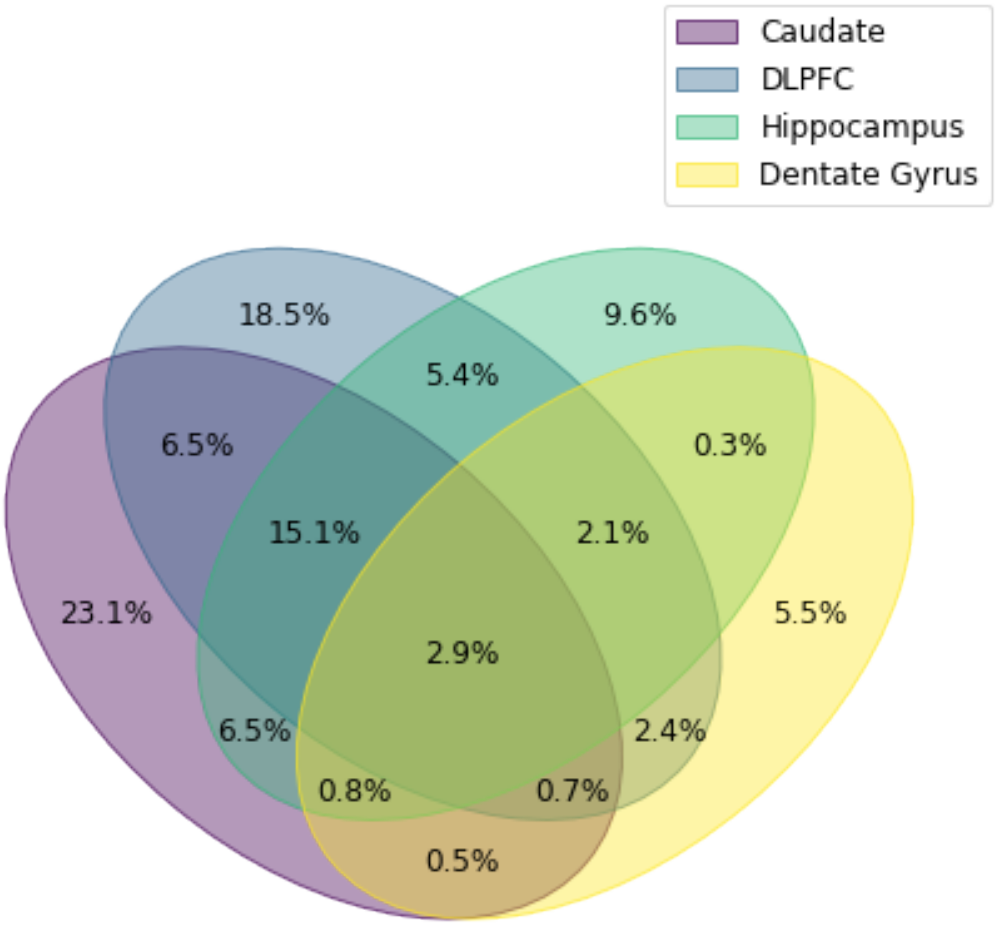

Caudate
DLPFC
Hippocampus
Dentate Gyrus
18.6%
10.0%
8.6%
7.1%
0.0%
18.6%
0.0%
25.7%
1.4%
5.7%
2.9%
0.0%
0.0%
0.0%
1.4%

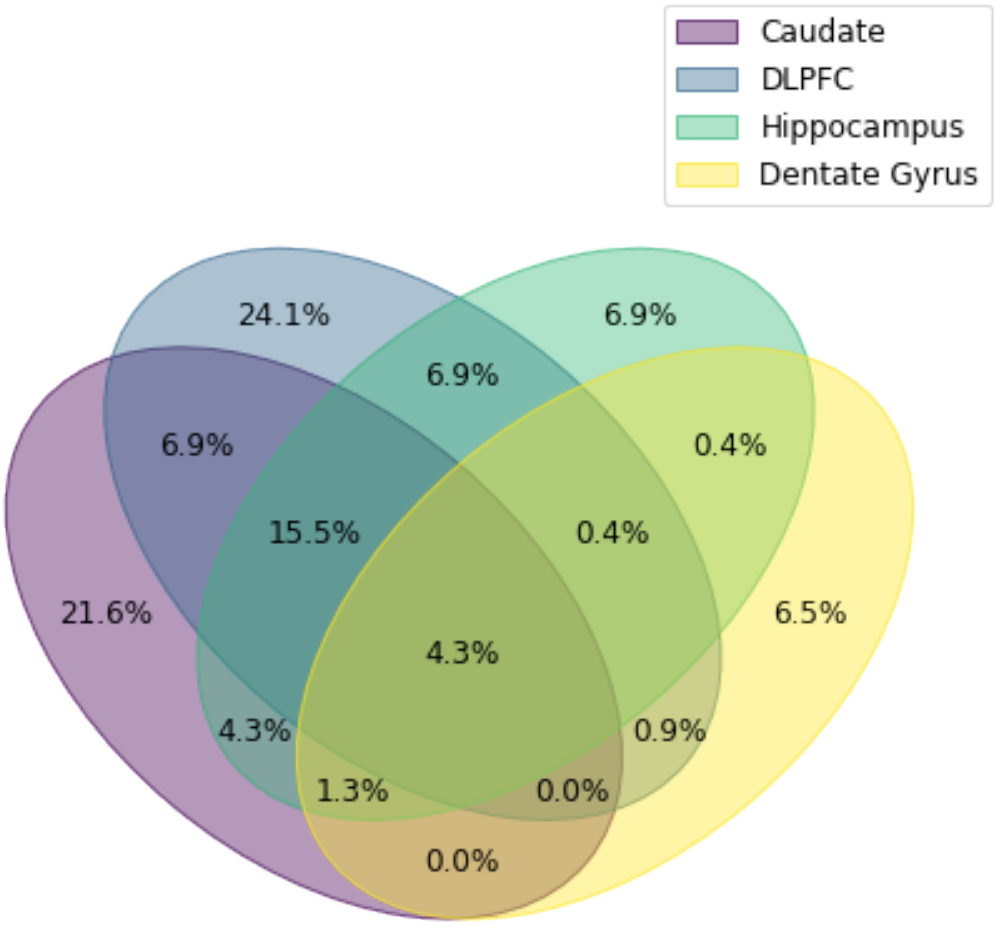

Caudate
DLPFC
Hippocampus
Dentate Gyrus
20.0%
16.7%
3.3%
13.3%
0.0%
20.0%
0.0%
13.3%
0.0%
6.7%
3.3%
0.0%
0.0%
0.0%
3.3%

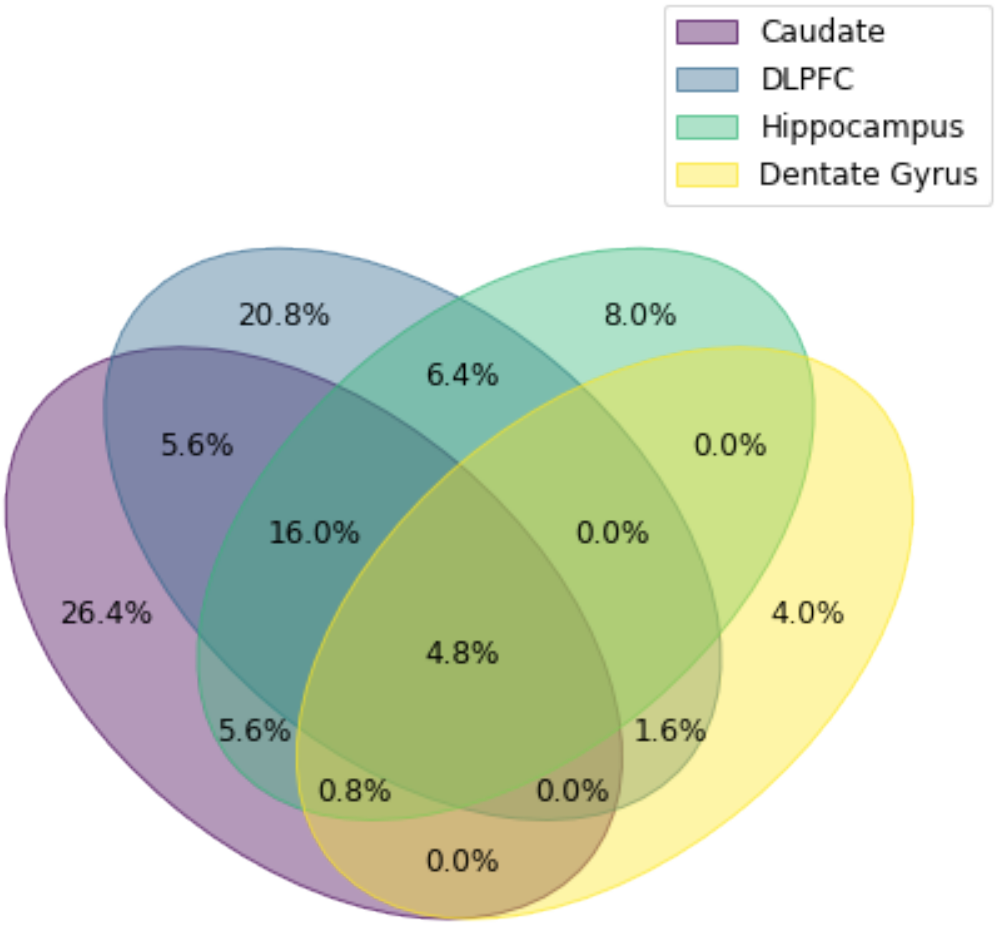

Caudate
DLPFC
Hippocampus
Dentate Gyrus
17.4%
26.1%
8.7%
4.3%
0.0%
13.0%
0.0%
21.7%
0.0%
0.0%
4.3%
0.0%
0.0%
4.3%
0.0%

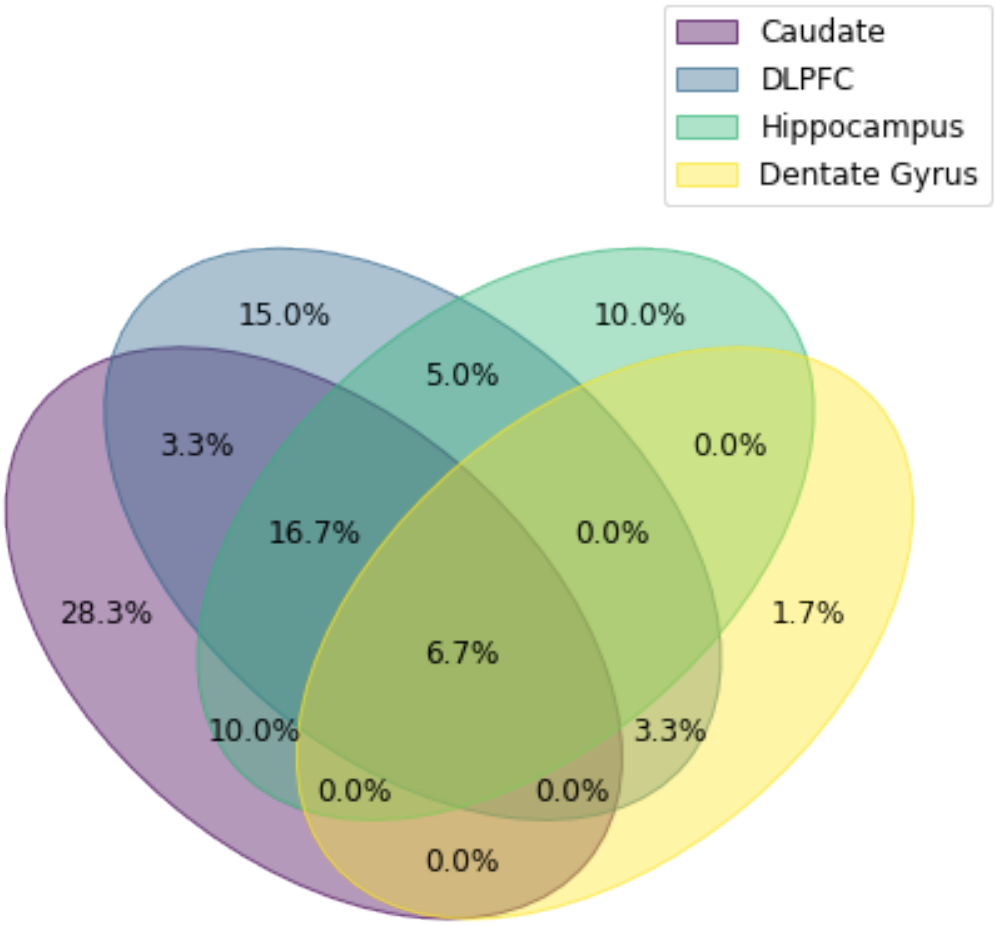

Caudate
DLPFC
Hippocampus
Dentate Gyrus
0.0%
33.3%
0.0%
0.0%
0.0%
11.1%
0.0%
22.2%
0.0%
0.0%
33.3%
0.0%
0.0%
0.0%
0.0%

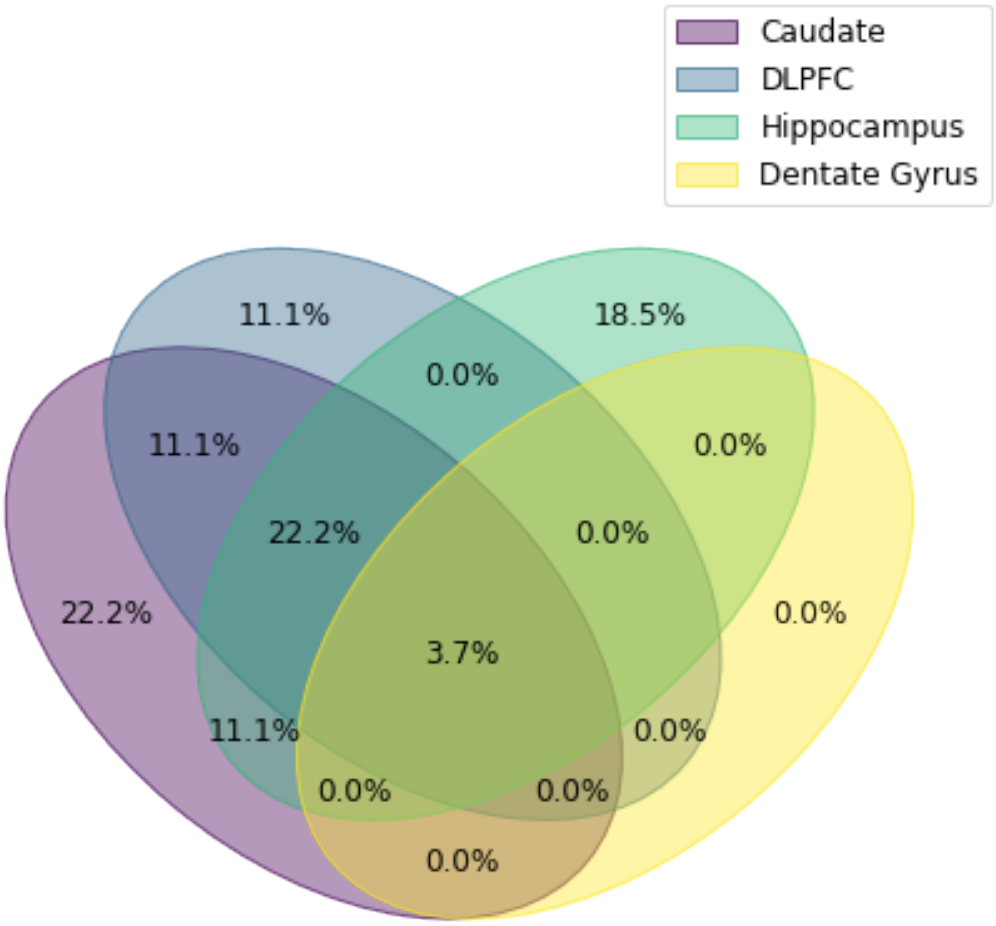

Caudate
DLPFC
Hippocampus
Dentate Gyrus
19.3%
15.0%
5.7%
6.2%
0.8%
12.7%
1.3%
24.2%
4.3%
2.8%
5.5%
0.9%
0.5%
0.4%
0.5%

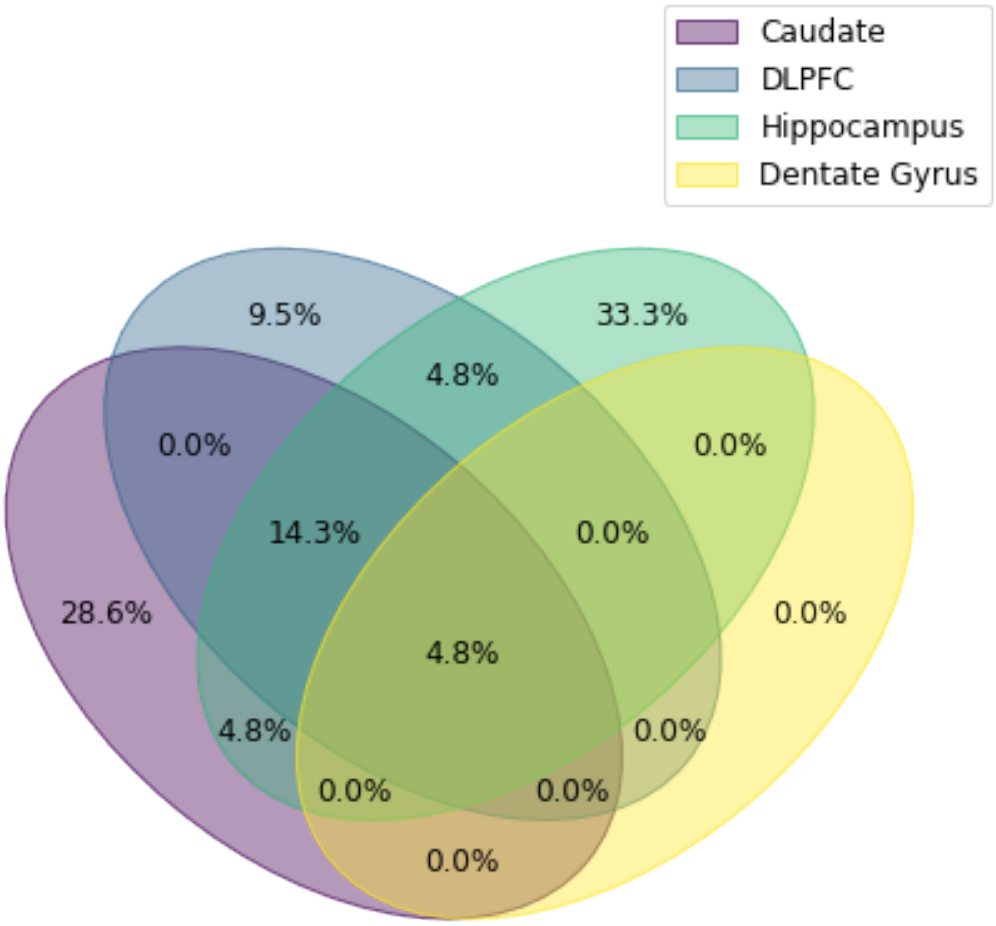

Caudate
DLPFC
Hippocampus
Dentate Gyrus
18.5%
9.6%
5.4%
6.5%
0.3%
15.1%
2.1%
23.1%
5.5%
2.9%
6.5%
2.4%
0.8%
0.7%
0.5%

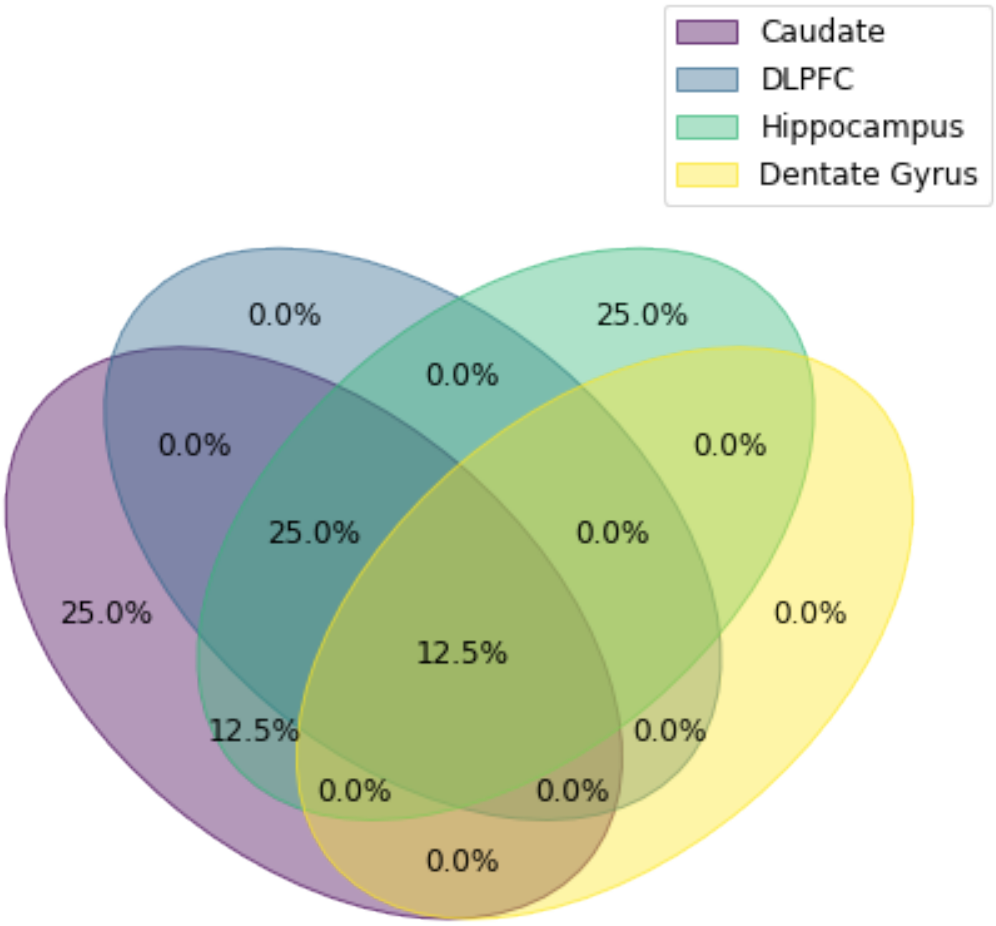

Caudate
DLPFC
Hippocampus
Dentate Gyrus
24.1%
6.9%
6.9%
6.9%
0.4%
15.5%
0.4%
21.6%
6.5%
4.3%
4.3%
0.9%
1.3%
0.0%
0.0%


Caudate
DLPFC
Hippocampus
Dentate Gyrus
20.8%
8.0%
6.4%
5.6%
0.0%
16.0%
0.0%
26.4%
4.0%
4.8%
5.6%
1.6%
0.8%
0.0%
0.0%


Caudate
DLPFC
Hippocampus
Dentate Gyrus
15.0%
10.0%
5.0%
3.3%
0.0%
16.7%
0.0%
28.3%
1.7%
6.7%
10.0%
3.3%
0.0%
0.0%
0.0%


Caudate
DLPFC
Hippocampus
Dentate Gyrus
11.1%
18.5%
0.0%
11.1%
0.0%
22.2%
0.0%
22.2%
0.0%
3.7%
11.1%
0.0%
0.0%
0.0%
0.0%


Caudate
DLPFC
Hippocampus
Dentate Gyrus
9.5%
33.3%
4.8%
0.0%
0.0%
14.3%
0.0%
28.6%
0.0%
4.8%
4.8%
0.0%
0.0%
0.0%
0.0%



1.4 Examining partial R2 results using most predictive SNPs

[18]:
```
partial.groupby("tissue").describe().T
```

[18]:

tissue		Caudate	DLPFC	Dentate Gyrus	Hippocampus
n_features	count	2867.000000	2660.000000	770.000000	2843.000000
	mean	12.514126	13.332331	14.385714	11.902919
	std	23.181572	27.876792	19.540510	22.613318
	min	1.000000	1.000000	1.000000	1.000000
	25%	3.000000	3.000000	4.000000	2.000000
	50%	4.000000	5.000000	8.000000	4.000000
	75%	13.000000	15.000000	15.750000	12.000000
	max	433.000000	915.000000	199.000000	347.000000

```
test_score_r2 count  2867.000000  2660.000000     770.000000  2843.000000
              mean      0.121152     0.129189       0.201115     0.103866
              std       0.149158     0.154981       0.185990     0.142715
              min       0.000000     0.000000       0.000000     0.000000
              25%       0.014847     0.019266       0.055788     0.010154
              50%       0.064597     0.070831       0.144500     0.047870
              75%       0.169860     0.185071       0.281533     0.136068
              max       0.890767     0.914657       1.000000     0.921536
```

[19]:
```
partial[(partial["test_score_r2"] > 0.88)]
```

[19]:

	tissue	feature	n_features	test_score_r2	Model
1288	Caudate	ENSG00000166435.15	25	0.890767	Partial R2
5210	DLPFC	ENSG00000257218.5	149	0.907953	Partial R2
5501	DLPFC	ENSG00000279672.1	146	0.914657	Partial R2
5565	Hippocampus	ENSG00000013573.16	9	0.880784	Partial R2
5996	Hippocampus	ENSG00000111788.10	84	0.882758	Partial R2
7888	Hippocampus	ENSG00000244879.5	166	0.908674	Partial R2
7996	Hippocampus	ENSG00000255374.3	74	0.921536	Partial R2
8392	Dentate Gyrus	ENSG00000065325.12	95	1.000000	Partial R2

- *GLP2R* (ENSG00000065325) Glucagon Like Peptide 2 Receptor

[20]:
```
idv_partial = pd.read_csv("../../partial_r2/individual_partial_r2_metrics.tsv",␣
↪sep='\t')
idv_partial.head(2)
```

[20]:

	SNP	Partial_R2	Full_R2	Reduced_R2	Tissue	Geneid
0	chrX_30633576_C_T_0	0.000065	227.834700	227.849408	Caudate	ENSG00000198814.12
1	chrX_30633576_C_T_1	0.008910	225.819164	227.849408	Caudate	ENSG00000198814.12

[21]:
```
idv_partial[["Partial_R2", "Tissue", "Geneid"]].groupby("Tissue").describe().T
```

[21]:
```
Tissue                Caudate         DLPFC  Dentate Gyrus   Hippocampus
Partial_R2 count  1.762851e+06  1.595825e+06  450379.000000  1.720189e+06
           mean   1.177298e-02  1.192422e-02       0.017215  1.016017e-02
           std    3.623054e-02  3.529084e-02       0.042975  3.167019e-02
           min    0.000000e+00  0.000000e+00       0.000000  0.000000e+00
           25%    0.000000e+00  0.000000e+00       0.000000  0.000000e+00
           50%    1.151334e-03  1.190616e-03       0.001776  1.026979e-03
           75%    8.549333e-03  8.895461e-03       0.016113  7.504191e-03
           max    8.853651e-01  9.086128e-01       0.927805  8.921231e-01
```

The vast majority of SNPs to not hold a lot of information (partial r2 < 0.01) with 25% close to 0.

[22]:
```
idv_partial.loc[(idv_partial["Partial_R2"] >= 0.8), ["Tissue", "Partial_R2",␣
↪"Geneid"]].groupby("Tissue").size()
```

[22]:
```
Tissue
Caudate          129
DLPFC             25
Dentate Gyrus     29
Hippocampus       78
dtype: int64
```

[23]:
```
idv_partial.loc[(idv_partial["Partial_R2"] >= 0.8), ["Tissue", "Partial_R2",␣
↪"Geneid"]].groupby("Geneid").size()
```

[23]:
```
Geneid
ENSG00000013573.16    93
ENSG00000074803.17     3
ENSG00000142856.16    19
ENSG00000164346.9     17
ENSG00000166435.15    12
ENSG00000228906.1     27
ENSG00000255374.3     58
ENSG00000256274.1      3
ENSG00000267370.1      3
ENSG00000270605.1     26
dtype: int64
```

[24]:
```
idv_partial.loc[(idv_partial["Partial_R2"] >= 0.8), ["Tissue", "Partial_R2",␣
↪"Geneid"]].groupby(["Geneid", "Tissue"]).size()
```

[24]:
```
Geneid              Tissue
ENSG00000013573.16  Caudate          93
ENSG00000074803.17  DLPFC             3
ENSG00000142856.16  Caudate           7
                    DLPFC             7
                    Hippocampus       5
ENSG00000164346.9   Caudate          17
ENSG00000166435.15  Caudate           3
                    DLPFC             3
                    Dentate Gyrus     3
                    Hippocampus       3
ENSG00000228906.1   Caudate           9
                    DLPFC             9
                    Hippocampus       9
ENSG00000255374.3   Hippocampus      58
ENSG00000256274.1   Hippocampus       3
ENSG00000267370.1   DLPFC             3
```

```
ENSG00000270605.1   Dentate Gyrus    26
dtype: int64
```

[ ]: