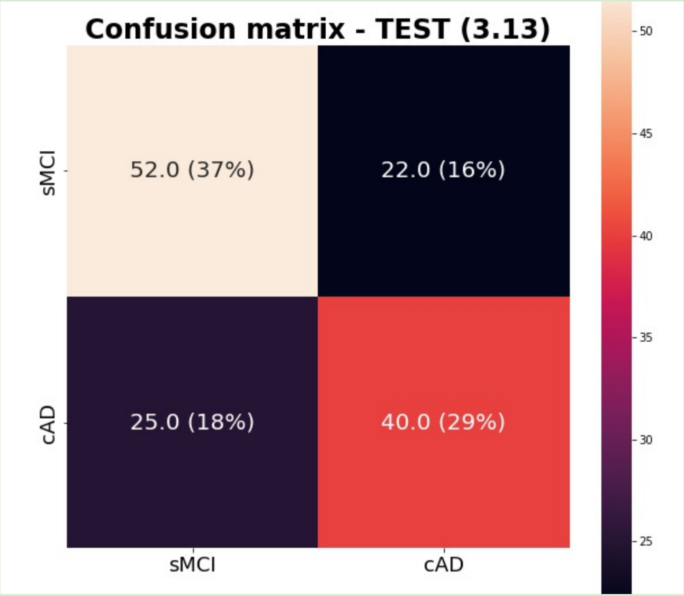
Sammenligning av de to modellenes performance:

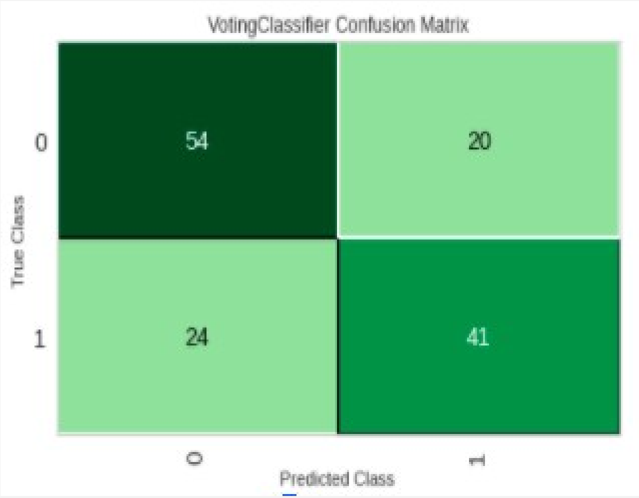
Mean across CV
Accuracy: 74.6
Precision: 73.6
Recall: 78.3

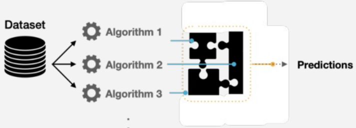
TEST
Accuracy: 66.2
Precision: 64.5
Recall: 61.5

Mean across CV
Accuracy: 74.85
Precision: 74.07
Recall: 75.40

TEST
Accuracy: 68.34
Precision: 66.67
Recall: 64.61







Ingrid/pycaret_K50/pycaret_accuracy_top5_K50.ipynb

FOLDS = 50
Select = top5
Sort = ‘Accuracy’

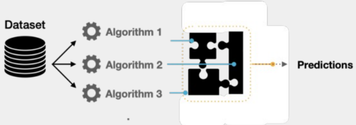
Vi prøver alle og plukker ut top 5:

```
[48]: top5 = compare_models(n_select=5, sort='Accuracy')
```

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
et	Extra Trees Classifier	0.7287	0.7878	0.7193	0.7336	0.7070	0.4555	0.4734	0.0390
lda	Linear Discriminant Analysis	0.7207	0.7951	0.7340	0.7166	0.7101	0.4425	0.4573	0.0020
ada	Ada Boost Classifier	0.7120	0.7384	0.7193	0.7153	0.7000	0.4239	0.4381	0.0152
lr	Logistic Regression	0.7065	0.7676	0.7267	0.6856	0.6949	0.4119	0.4190	0.0600
nb	Naive Bayes	0.7027	0.7528	0.7700	0.6865	0.7085	0.4085	0.4304	0.0018
ridge	Ridge Classifier	0.7025	0.0000	0.7420	0.6931	0.7008	0.4061	0.4188	0.0018
rf	Random Forest Classifier	0.6956	0.7800	0.6907	0.6929	0.6736	0.3896	0.4046	0.0492
lightgbm	Light Gradient Boosting Machine	0.6822	0.7545	0.6753	0.6677	0.6585	0.3624	0.3732	0.0190
qda	Quadratic Discriminant Analysis	0.6805	0.7480	0.6740	0.6648	0.6537	0.3571	0.3735	0.0020
gbc	Gradient Boosting Classifier	0.6778	0.7371	0.6960	0.6605	0.6610	0.3554	0.3695	0.0186
dt	Decision Tree Classifier	0.6362	0.6350	0.6313	0.6377	0.6098	0.2703	0.2883	0.0022
knn	K Neighbors Classifier	0.6245	0.6460	0.6633	0.6124	0.6129	0.2489	0.2668	0.0038
svm	SVM - Linear Kernel	0.5960	0.0000	0.5627	0.5043	0.4777	0.1839	0.2285	0.0022
dummy	Dummy Classifier	0.5245	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0018

```
[51]: tuned_top5[0]
```

```
[51]: ExtraTreesClassifier(bootstrap=True, ccp_alpha=0.0, class_weight='balanced',
                           criterion='entropy', max_depth=6, max_features=1.0,
                           max_leaf_nodes=None, max_samples=None,
                           min_impurity_decrease=0.0001, min_impurity_split=None,
                           min_samples_leaf=2, min_samples_split=9,
                           min_weight_fraction_leaf=0.0, n_estimators=100, n_jobs=-1,
                           oob_score=False, random_state=8061, verbose=0,
                           warm_start=False)
```



Ingrid/pycaret_K50/
pycaret_accuracy_top5_K50.ipynb

FOLDS = 50, Select = top5,
Sort = ‘Accuracy’

Vi prøver alle og plukker ut top 5:

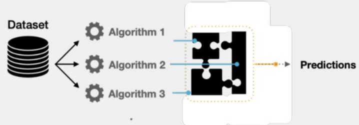
```
[96]: top5 = compare_models(n_select=5, sort='Accuracy')
```

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
et	Extra Trees Classifier	0.7336	0.7877	0.7260	0.7403	0.7115	0.4653	0.4859	0.0394
rf	Random Forest Classifier	0.7222	0.7872	0.7387	0.7159	0.7093	0.4441	0.4606	0.0462
lda	Linear Discriminant Analysis	0.7207	0.7951	0.7340	0.7166	0.7101	0.4425	0.4573	0.0018
ada	Ada Boost Classifier	0.7120	0.7384	0.7193	0.7153	0.7000	0.4239	0.4381	0.0152
lr	Logistic Regression	0.7065	0.7676	0.7267	0.6856	0.6949	0.4119	0.4190	0.0564
nb	Naive Bayes	0.7027	0.7528	0.7700	0.6865	0.7085	0.4085	0.4304	0.0018
ridge	Ridge Classifier	0.7025	0.0000	0.7420	0.6931	0.7008	0.4061	0.4188	0.0018
lightgbm	Light Gradient Boosting Machine	0.6822	0.7545	0.6753	0.6677	0.6585	0.3624	0.3732	0.0184
qda	Quadratic Discriminant Analysis	0.6805	0.7480	0.6740	0.6648	0.6537	0.3571	0.3735	0.0020
gbc	Gradient Boosting Classifier	0.6724	0.7323	0.6920	0.6541	0.6566	0.3439	0.3566	0.0182
dt	Decision Tree Classifier	0.6380	0.6373	0.6207	0.6325	0.6068	0.2751	0.2915	0.0020
knn	K Neighbors Classifier	0.6245	0.6460	0.6633	0.6124	0.6129	0.2489	0.2668	0.0036
svm	SVM - Linear Kernel	0.6102	0.0000	0.5687	0.4549	0.4733	0.2084	0.2413	0.0020
dummy	Dummy Classifier	0.5245	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0016

Top5

```
[97]: top5
```

```
[97]: [ExtraTreesClassifier(bootstrap=False, ccp_alpha=0.0, class_weight=None,
                           criterion='gini', max_depth=None, max_features='auto',
                           max_leaf_nodes=None, max_samples=None,
                           min_impurity_decrease=0.0, min_impurity_split=None,
                           min_samples_leaf=1, min_samples_split=2,
                           min_weight_fraction_leaf=0.0, n_estimators=100, n_jobs=-1,
                           oob_score=False, random_state=1138, verbose=0,
                           warm_start=False),
      RandomForestClassifier(bootstrap=True, ccp_alpha=0.0, class_weight=None,
                             criterion='gini', max_depth=None, max_features='auto',
                             max_leaf_nodes=None, max_samples=None,
                             min_impurity_decrease=0.0, min_impurity_split=None,
                             min_samples_leaf=1, min_samples_split=2,
                             min_weight_fraction_leaf=0.0, n_estimators=100,
                             n_jobs=-1, oob_score=False, random_state=1138, verbose=0,
                             warm_start=False),
      LinearDiscriminantAnalysis(n_components=None, priors=None, shrinkage=None,
                                 solver='svd', store_covariance=False, tol=0.0001),
      AdaBoostClassifier(algorithm='SAMME.R', base_estimator=None, learning_rate=1.0,
                         n_estimators=50, random_state=1138),
      LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
                         intercept_scaling=1, l1_ratio=None, max_iter=1000,
                         multi_class='auto', n_jobs=None, penalty='l2',
                         random_state=1138, solver='lbfgs', tol=0.0001, verbose=0,
                         warm_start=False)]
```



tuned_top5

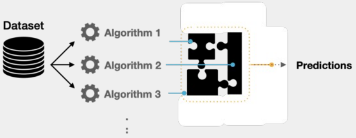
```
[98]: tuned_top5 = [tune_model(i) for i in top5]
```

	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
0	0.7273	0.8000	0.8333	0.7143	0.7692	0.4407	0.4485
1	0.7273	0.8000	0.8333	0.7143	0.7692	0.4407	0.4485
2	0.5455	0.6000	0.6667	0.5714	0.6154	0.0678	0.0690
3	0.8182	0.9000	1.0000	0.7500	0.8571	0.6207	0.6708
4	0.7273	0.7333	0.6667	0.8000	0.7273	0.4590	0.4667
5	0.3636	0.2667	0.3333	0.4000	0.3636	-0.2623	-0.2667
6	0.8182	0.8667	1.0000	0.7143	0.8333	0.6452	0.6901

Her er fem bagging ensembles av alle top 5:

```
[99]: tuned_top5
```

```
[99]: [ExtraTreesClassifier(bootstrap=False, ccp_alpha=0.0,
                           class_weight='balanced_subsample', criterion='entropy',
                           max_depth=10, max_features=1.0, max_leaf_nodes=None,
                           max_samples=None, min_impurity_decrease=0.002,
                           min_impurity_split=None, min_samples_leaf=2,
                           min_samples_split=5, min_weight_fraction_leaf=0.0,
                           n_estimators=260, n_jobs=-1, oob_score=False,
                           random_state=1138, verbose=0, warm_start=False),
 RandomForestClassifier(bootstrap=True, ccp_alpha=0.0, class_weight={},
                         criterion='gini', max_depth=11, max_features='log2',
                         max_leaf_nodes=None, max_samples=None,
                         min_impurity_decrease=0.0001, min_impurity_split=None,
                         min_samples_leaf=6, min_samples_split=2,
                         min_weight_fraction_leaf=0.0, n_estimators=190,
                         n_jobs=-1, oob_score=False, random_state=1138, verbose=0
                         warm_start=False),
 LinearDiscriminantAnalysis(n_components=None, priors=None, shrinkage='auto',
                             solver='lsqr', store_covariance=False, tol=0.0001),
 AdaBoostClassifier(algorithm='SAMME', base_estimator=None, learning_rate=0.05,
                     n_estimators=200, random_state=1138),
 LogisticRegression(C=0.593, class_weight={}, dual=False, fit_intercept=True,
                     intercept_scaling=1, l1_ratio=None, max_iter=1000,
                     multi_class='auto', n_jobs=None, penalty='l2',
                     random_state=1138, solver='lbfgs', tol=0.0001, verbose=0,
                     warm_start=False)]
```



bagged_top5 = [ensemble_model(i) for i in tuned_top5]

Dette er et ensemble av top5 tuned

Ensembling a trained model is as simple as writing **ensemble_model.** It takes only one mandatory parameter i.e. the trained model object. This functions returns a table with k-fold cross validated scores of common evaluation metrics along with trained model object. The evaluation metrics used are:

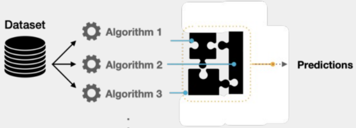
49	0.7000	0.6800	0.8000	0.6667	0.7273	0.4000	0.4082
Mean	0.7065	0.7629	0.7067	0.6983	0.6888	0.4105	0.4189
SD	0.1562	0.1676	0.2030	0.1969	0.1813	0.3114	0.3213

bagged_top5 = [ensemble_model(i) for i in top5]

Dette er et ensemble av top5 ikke-tuned

Ensembling a trained model is as simple as writing **ensemble_model.** It takes only one mandatory parameter i.e. the trained model object. This functions returns a table with k-fold cross validated scores of common evaluation metrics along with trained model object. The evaluation metrics used are:

48	0.6000	0.5200	0.6000	0.6000	0.6000	0.2000	0.2000
49	0.7000	0.6800	0.8000	0.6667	0.7273	0.4000	0.4082
Mean	0.7065	0.7636	0.7067	0.6997	0.6883	0.4105	0.4201
SD	0.1519	0.1682	0.2030	0.1952	0.1776	0.3029	0.3128



nontuned_blender = blend_models(estimator_list = top5)

Dette er et ensemble av top5 ikke-tuned

Blending models is a method of ensembling which uses consensus among estimators to generate final predictions. The idea behind blending is to combine different machine learning algorithms and use a majority vote or the average predicted probabilities in case of classification to predict the final outcome. NB! Her benytter we VotingClassifier - dette må være noe innebygget for ser ikke ut som Alexander spesifierer dette noe sted…

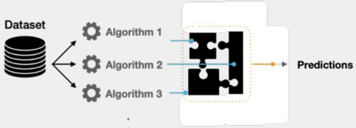
49	0.7000	0.6800	0.6000	0.7500	0.6667	0.4000	0.4082
Mean	0.7356	0.7945	0.7500	0.7217	0.7230	0.4705	0.4832
SD	0.1584	0.1707	0.2121	0.1990	0.1815	0.3159	0.3250

tuned_blender = blend_models(estimator_list = top5)

Dette er et ensemble av top5 tuned

Blending models is a method of ensembling which uses consensus among estimators to generate final predictions. The idea behind blending is to combine different machine learning algorithms and use a majority vote or the average predicted probabilities in case of classification to predict the final outcome. NB! Her benytter we VotingClassifier - dette må være noe innebygget for ser ikke ut som Alexander spesifierer dette noe sted…

49	0.5000	0.6400	0.4000	0.5000	0.4444	0.0000	0.0000
Mean	0.7351	0.7987	0.7540	0.7263	0.7251	0.4697	0.4839
SD	0.1686	0.1659	0.2186	0.2050	0.1874	0.3371	0.3449



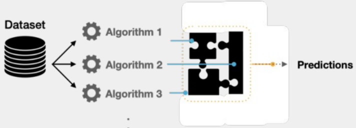
The winner (accodring to accuracy) is:

bagged_blender = blend_models(estimator_list = bagged_top5)

Dette er et ensemble av top5 bagged_top5

Blending models is a method of ensembling which uses consensus among estimators to generate final predictions. The idea behind blending is to combine different machine learning algorithms and use a majority vote or the average predicted probabilities in case of classification to predict the final outcome. NB! Her benytter we VotingClassifier - dette må være noe innebygget for ser ikke ut som Alexander spesifierer dette noe sted…

49	0.7000	0.6800	0.6000	0.7500	0.6667	0.4000	0.4082
Mean	0.7485	0.7999	0.7540	0.7407	0.7315	0.4962	0.5113
SD	0.1463	0.1654	0.2149	0.1947	0.1755	0.2924	0.3022



Undersøker overlapp i de to modellenes misklassifikasjoner - sMCI

Vi ser her at modellene missklassifiserte 18 av de samme sMCI-subjektene som konvertitter.

[149]:
```
fpteller = 0
for i in final_df.index:
    if final_df.loc[i,'Ens_pred'] == 'FP_' and final_df.loc[i,'CM_pred_'] == 'FP':
        fpteller += 1
print("*"*90)
print(f"Random Forest: feilklassifiserte {FP_teller} sMCI som konvertitter.")
print(f"Ensemblet: feilklassifiserte {fp_teller} sMCI som konvertitter.")
print()
print(f"Av disse feilklassifiseringene overlappet modellene på {fpteller} deltagere.")
print("*"*90)
```

```
******************************************************************************************
Random Forest: feilklassifiserte 22 sMCI som konvertitter.
Ensemblet: feilklassifiserte 20 sMCI som konvertitter.

Av disse feilklassifiseringene overlappet modellene på 18 deltagere.
******************************************************************************************
```

Identifiserer index for de som ble feilklassifisert ulikt (sMCI --> FP)

Pasienter som ble klassifisert som FP aav ensemble og IKKE FP (altså nødvendigvis TN) av Random Forst.

Med andre ord: stabile pasienter som av ensemblet ble klassifisert som konvertitter men som av Random Forest ble klassifisert som stabile

[145]:
```
indeksList = []
for i in final_df.index:
    if final_df.loc[i,'Ens_pred'] == 'FP_' and final_df.loc[i,'CM_pred_'] == 'TN':
        indeksList.append(i)

print("*"*150)
print(f"Pasienter med {len(indeksList)} følgende indekser {indeksList} ble korrekt klassifisert av RF, men feilaktig klassifisert av ensemblet")
print("*"*150)
```

```
******************************************************************************************************************************************************
Pasienter med 2 følgende indekser [5565, 6309] ble korrekt klassifisert av RF, men feilaktig klassifisert av ensemblet
******************************************************************************************************************************************************
```

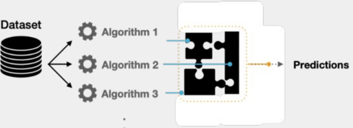
Pasienter som ble klassifisert som IKKE FP (altså nødvendigvis TN) av ensembelet og som FP av Random Foerst

Med andre ord: korrekt klassifisert av ensemblet og feilklassifisert i RF

[133]:
```
indeksList = []
for i in final_df.index:
    if final_df.loc[i,'Ens_pred'] == 'TN_' and final_df.loc[i,'CM_pred_'] == 'FP':
        indeksList.append(i)

print("*"*150)
print(f"Pasienter med {len(indeksList)} følgende indexer {indeksList} ble korrekt klassifisert av ensemblet, men feilaktig klassifisert av Random Forest")
print("*"*150)
```

```
******************************************************************************************************************************************************
Pasienter med 4 følgende indexer [1203, 2001, 2567, 4989] ble korrekt klassifisert av ensemblet, men feilaktig klassifisert av Random Forest
******************************************************************************************************************************************************
```



Undersøker overlapp i de to modellenes misklassifikasjonener - cAD

Vi ser her at modellene missklassifiserte 20 av de samme cAD-subjektene som stabile.

[51]:
```
# Dette viser at de to modellene feilklassifiserte 19 av de subjects som konvertitter
fnteller = 0
for i in final_df.index:
    if final_df.loc[i,'Ens_pred'] == 'FN_' and final_df.loc[i,'CM_pred_'] == 'FN':
        fnteller += 1

print("*"*90)
print(f"Random Forest: feilklassifiserte {FN_teller} cAD som stabile.")
print(f"Ensemblet: feilklassifiserte {fn_teller} cAD som stabile.")
print()
print(f"Av disse feilklassifiseringene overlappet de to modellene på {fnteller} deltagere.")
print("*"*90)
```

```
******************************************************************************************
Random Forest: feilklassifiserte 25 cAD som stabile.
Ensemblet: feilklassifiserte 24 cAD som stabile.

Av disse feilklassifiseringene overlappet de to modellene på 20 deltagere.
******************************************************************************************
```

Identifiserer index for de som ble feilklassifisert ulikt (cAD --> FN)

Pasienter som ble korrekt klassifisert av Random Forest, men feilaktig klassifisert av ensemblet.

[52]:
```
indeksList = []
for i in final_df.index:
    if final_df.loc[i,'Ens_pred'] == 'FN_' and final_df.loc[i,'CM_pred_'] == 'TP':
        indeksList.append(i)

print("*"*150)
print(f"Pasienter med {len(indeksList)} følgende indexer {indeksList} ble korrekt klassifisert av RF, men feilaktig klassifisert av ensemblet")
print("*"*150)
```

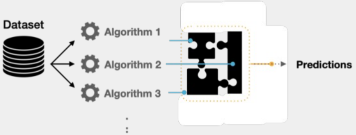
```
******************************************************************************************************************************************************
Pasienter med 4 følgende indexer [384, 899, 1727, 5309] ble korrekt klassifisert av RF, men feilaktig klassifisert av ensemblet
******************************************************************************************************************************************************
```

Pasienter som ble korrekt klassifisert av ensemblet og feilaktig klassifisert av Random Forest.

[53]:
```
indeksList = []
for i in final_df.index:
    if final_df.loc[i,'Ens_pred'] == 'TP_' and final_df.loc[i,'CM_pred_'] == 'FN':
        indeksList.append(i)

print("*"*150)
print(f"Pasienter med {len(indeksList)} følgende indexer {indeksList} ble korrekt klassifisert av ensemblet, men feilaktig klassifisert av RF")
print("*"*150)
```

```
******************************************************************************************************************************************************
Pasienter med 5 følgende indexer [1355, 2119, 3080, 3177, 4300] ble korrekt klassifisert av ensemblet, men feilaktig klassifisert av RF
******************************************************************************************************************************************************
```



Comparative statistics for correct vs incorrect classified sMCI (demographics)

“**True” stable MCI** who were **wrongfully classified as converting to AD** (TN vs FP) had**:**

- Higher age
- No difference in education or gender

Contingency Tables

CM_pred_		PTGENDER Female	Male	Total
FP	Observed	8	16	24
	Expected	10.4	13.6	24.0
TN	Observed	24	26	50
	Expected	21.6	28.4	50.0
Total	Observed	32	42	74
	Expected	32.0	42.0	74.0

χ² Tests

	Value	df	p
χ²	1.42	1	0.233
N	74		

Nominal

	Value
Phi-coefficient	0.139
Cramer's V	0.139

Independent Samples T-Test

		statistic	df	p	Cohen's d
AGE	Student's t	3.63	72.0	<.001	0.900
	Mann-Whitney U	299		<.001	0.900
PTEDUCAT	Student's t	1.05	72.0	0.295	0.262
	Mann-Whitney U	483		0.173	0.262

Group Descriptives

	Group	N	Mean	Median	SD	SE
AGE	FP	24	77.0	77.1	5.31	1.084
	TN	50	70.9	71.6	7.39	1.045
PTEDUCAT	FP	24	16.8	18.0	3.40	0.694
	TN	50	16.0	16.0	2.83	0.400

Comparative statistics for correct vs incorrect classified sMCI (cog. feat. used in model)

“**True” stable MCI** who were **wrongfully classified as converting to AD** (TN vs FP) had**:**

- Worse score on all memory tests
- Worse on TMT B
- Smaller hippocampus

Group Descriptives

	Group	N	Mean	Median	SD	SE
RAVLT_immediate	FP	24	27.91667	29.00000	4.61	0.942
	TN	50	41.12000	40.00000	7.68	1.087
AVDEL30MIN_neuro	FP	24	1.58333	1.00000	1.69	0.345
	TN	50	6.08000	6.00000	3.47	0.491
AVDELTOT_neuro	FP	24	8.79167	9.00000	3.37	0.689
	TN	50	12.66000	13.00000	2.04	0.288
TRAASCOR_neuro	FP	24	40.83333	41.00000	9.81	2.002
	TN	50	36.86000	33.00000	12.89	1.823
TRABSCOR_neuro	FP	24	124.83333	113.50000	48.81	9.963
	TN	50	90.88000	82.00000	38.57	5.454
CATANIMSC_neuro	FP	24	16.70833	16.50000	5.63	1.149
	TN	50	18.88000	18.50000	4.57	0.646
LRHHC_n_long	FP	24	0.00387	0.00401	6.08e–4	1.24e–4
	TN	50	0.00462	0.00459	7.32e–4	1.04e–4
ANARTERR_neuro	FP	24	10.95833	10.00000	8.04	1.640
	TN	50	12.46000	9.50000	9.78	1.383
GDTOTAL_gds	FP	24	1.66667	2.00000	1.31	0.267
	TN	50	1.90000	2.00000	1.28	0.181

Independent Samples T-Test

		statistic	df	p	Cohen's d
RAVLT_immediate	Student's t	–7.757 [a]	72.0	< .001	–1.926
	Mann-Whitney U	67.0		< .001	–1.926
AVDEL30MIN_neuro	Student's t	–6.001 [a]	72.0	< .001	–1.490
	Mann-Whitney U	141.0		< .001	–1.490
AVDELTOT_neuro	Student's t	–6.129 [a]	72.0	< .001	–1.522
	Mann-Whitney U	205.0		< .001	–1.522
TRAASCOR_neuro	Student's t	1.334	72.0	0.186	0.331
	Mann-Whitney U	419.5		0.037	0.331
TRABSCOR_neuro	Student's t	3.247 [a]	72.0	0.002	0.806
	Mann-Whitney U	319.0		0.001	0.806
CATANIMSC_neuro	Student's t	–1.773	72.0	0.081	–0.440
	Mann-Whitney U	449.0		0.081	–0.440
LRHHC_n_long	Student's t	–4.331	72.0	< .001	–1.075
	Mann-Whitney U	262.0		< .001	–1.075
ANARTERR_neuro	Student's t	–0.653	72.0	0.516	–0.162
	Mann-Whitney U	571.0		0.742	–0.162
GDTOTAL_gds	Student's t	–0.728	72.0	0.469	–0.181
	Mann-Whitney U	539.0		0.471	–0.181

Comparative statistics for correct vs incorrect classified sMCI (biological feat.)

“**True” stable MCI** who were **wrongfully classified as converting to AD** (TN vs FP) had**:**

- Smaller hippocampus
- No difference in ApoE

Contingency Tables

CM_pred_		Apoe4_ 0	1	Total
FP	Observed	16	8	24
	Expected	13.9	10.1	24.0
TN	Observed	27	23	50
	Expected	29.1	20.9	50.0
Total	Observed	43	31	74
	Expected	43.0	31.0	74.0

χ^2 Tests

	Value	df	p
χ^2	1.07	1	0.301
N	74		

Nominal

	Value
Phi-coefficient	0.120
Cramer's V	0.120

Independent Samples T-Test

		statistic	df	p	Cohen's d
LRHHC_n_long	Student's t	−4.33	72.0	<.001	−1.08
	Mann-Whitney U	262		<.001	−1.08

Group Descriptives

	Group	N	Mean	Median	SD	SE
LRHHC_n_long	FP	24	0.00387	0.00401	6.08e−4	1.24e−4
	TN	50	0.00462	0.00459	7.32e−4	1.04e−4

Comparative statistics for correct vs incorrect classified sMCI (diagnostics)

“**True” stable MCI** who were **wrongfully classified as converting to AD** (TN vs FP) had**:**

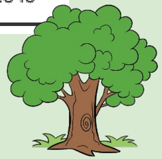
- Worse (lower) score on MMSE only

Independent Samples T-Test

		statistic	df	p	Cohen's d
MMSE	Student's t	–3.198	72.0	0.002	–0.794
	Mann-Whitney U	330		0.002	–0.794
CDRSB	Student's t	–1.142	72.0	0.257	–0.284
	Mann-Whitney U	525		0.369	–0.284
LDELTOTAL	Student's t	–0.869	72.0	0.388	–0.216
	Mann-Whitney U	504		0.267	–0.216
FAQTOTAL_faq	Student's t	0.540	72.0	0.591	0.134
	Mann-Whitney U	590		0.908	0.134

Group Descriptives

	Group	N	Mean	Median	SD	SE
MMSE	FP	24	26.96	27.000	1.681	0.343
	TN	50	28.30	29.00	1.693	0.239
CDRSB	FP	24	1.10	1.000	0.551	0.113
	TN	50	1.31	1.00	0.795	0.112
LDELTOTAL	FP	24	6.38	7.000	2.584	0.528
	TN	50	6.96	8.00	2.770	0.392
FAQTOTAL_faq	FP	24	2.71	0.500	4.506	0.920
	TN	50	2.16	1.00	3.883	0.549



Comparative statistics for correct vs incorrect classified cAD (demographics)

“True” cAD who were **wrongfully classified as stable** (FN vs TP) had**:**
- No difference in any demographics

Contingency Tables

CM_pred_		PTGENDER Female	Male	Total
FN	Observed	9	14	23
	Expected	8.85	14.2	23.0
TP	Observed	16	26	42
	Expected	16.15	25.8	42.0
Total	Observed	25	40	65
	Expected	25.00	40.0	65.0

χ^2 Tests

	Value	df	p
χ^2	0.00673	1	0.935
N	65		

Nominal

	Value
Phi-coefficient	0.0102
Cramer's V	0.0102

Independent Samples T-Test

		statistic	df	p	Cohen's d
AGE	Student's t	−0.811	63.0	0.420	−0.2104
	Mann-Whitney U	426		0.434	−0.2104
PTEDUCAT	Student's t	−0.307	63.0	0.760	−0.0797
	Mann-Whitney U	455		0.703	−0.0797

Group Descriptives

	Group	N	Mean	Median	SD	SE
AGE	FN	23	72.8	73.0	9.33	1.946
	TP	42	74.5	74.5	6.75	1.041
PTEDUCAT	FN	23	15.6	16.0	2.63	0.547
	TP	42	15.8	16.0	2.84	0.438

Comparative statistics for correct vs incorrect classified cAD (cog. feat. used in model)

“**True” cAD** who were **wrongfully classified as stable** (FN vs TP) had:

- Better scores on all three memory tests

Group Descriptives

	Group	N	Mean	Median	SD	SE
RAVLT_immediate	FN	23	35.87	37.00	7.44	1.551
	TP	42	28.02	29.50	4.75	0.733
AVDEL30MIN_neuro	FN	23	4.26	3.00	2.94	0.613
	TP	42	1.38	1.00	1.83	0.283
AVDELTOT_neuro	FN	23	11.52	12.00	2.68	0.558
	TP	42	8.83	10.00	3.69	0.569
TRAASCOR_neuro	FN	23	47.26	36.00	33.34	6.951
	TP	42	42.52	36.50	22.66	3.497
TRABSCOR_neuro	FN	23	133.87	96.00	85.55	17.838
	TP	42	132.19	111.50	79.56	12.276
CATANIMSC_neuro	FN	23	15.83	15.00	4.28	0.893
	TP	42	15.90	16.00	4.21	0.650
ANARTERR_neuro	FN	23	12.96	9.00	10.07	2.100
	TP	42	13.29	9.00	10.21	1.575
GDTOTAL_gds	FN	23	1.61	1.00	1.20	0.249
	TP	42	1.33	1.00	1.20	0.186

Independent Samples T-Test

		statistic	df	p	Cohen's d
RAVLT_immediate	Student's t	5.1876 [a]	63.0	< .001	1.3457
	Mann-Whitney U	194		< .001	1.3457
AVDEL30MIN_neuro	Student's t	4.8632 [a]	63.0	< .001	1.2615
	Mann-Whitney U	183		< .001	1.2615
AVDELTOT_neuro	Student's t	3.0752	63.0	0.003	0.7977
	Mann-Whitney U	278		0.005	0.7977
TRAASCOR_neuro	Student's t	0.6795	63.0	0.499	0.1763
	Mann-Whitney U	481		0.978	0.1763
TRABSCOR_neuro	Student's t	0.0792	63.0	0.937	0.0206
	Mann-Whitney U	458		0.731	0.0206
CATANIMSC_neuro	Student's t	−0.0716	63.0	0.943	−0.0186
	Mann-Whitney U	452		0.675	−0.0186
ANARTERR_neuro	Student's t	−0.1249	63.0	0.901	−0.0324
	Mann-Whitney U	466		0.821	−0.0324
GDTOTAL_gds	Student's t	0.8842	63.0	0.380	0.2294
	Mann-Whitney U	411		0.301	0.2294

Comparative statistics for correct vs incorrect classified cAD (biological markers)

“**True” cAD** who were **wrongfully classified as stable** (FN vs TP) had**:**

- Smaller hippocampus
- No difference in ApoE

Contingency Tables

		Apoe4_		
CM_pred_		0	1	Total
FN	Observed	11	12	23
	Expected	7.78	15.2	23.0
TP	Observed	11	31	42
	Expected	14.22	27.8	42.0
Total	Observed	22	43	65
	Expected	22.00	43.0	65.0

χ^2 Tests

	Value	df	p
χ^2	3.11	1	0.078
N	65		

Nominal

	Value
Phi-coefficient	0.219
Cramer's V	0.219

Independent Samples T-Test

		statistic	df	p	Cohen's d
LRHHC_n_long	Student's t	6.36	63.0	<.001	1.65
	Mann-Whitney U	131		<.001	1.65

Group Descriptives

	Group	N	Mean	Median	SD	SE
LRHHC_n_long	FN	23	0.00458	0.00455	6.56e–4	1.37e–4
	TP	42	0.00364	0.00366	5.25e–4	8.10e–5

Comparative statistics for correct vs incorrect classified cAD (diagnostics)

“True” cAD who were **wrongfully classified as stable** (FN vs TP) had**:**

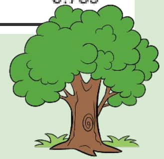
- Worse (higher or lower according to scale) on all except MMSE
  - NB! The “opposite” of wrongfully classified sMCI

Independent Samples T-Test

		statistic	df	p	Cohen's d
MMSE	Student's t	1.31	63.0	0.196	0.339
	Mann-Whitney U	388		0.185	0.339
CDRSB	Student's t	−2.49	63.0	0.016	−0.645
	Mann-Whitney U	312		0.017	−0.645
LDELTOTAL	Student's t	3.83	63.0	<.001	0.995
	Mann-Whitney U	229		<.001	0.995
FAQTOTAL_faq	Student's t	−2.08	63.0	0.042	−0.539
	Mann-Whitney U	315		0.021	−0.539

Group Descriptives

	Group	N	Mean	Median	SD	SE
MMSE	FN	23	27.22	28.00	1.882	0.392
	TP	42	26.62	27.00	1.696	0.262
CDRSB	FN	23	1.59	2.00	0.685	0.143
	TP	42	2.13	2.25	0.918	0.142
LDELTOTAL	FN	23	5.78	6.00	3.089	0.644
	TP	42	2.88	2.00	2.822	0.435
FAQTOTAL_faq	FN	23	2.83	2.00	3.822	0.797
	TP	42	5.24	4.00	4.787	0.739



Comparative statistics for correct vs incorrect classified sMCI (demographics)

“**True” stable MCI** who were **wrongfully classified as converting to AD** (TN vs FP) had**:**

- No difference in any demographics

Contingency Tables

Ens_pred		PTGENDER Female	Male	Total
FP_	Observed	12	9	21
	Expected	9.08	11.9	21.0
TN_	Observed	20	33	53
	Expected	22.92	30.1	53.0
Total	Observed	32	42	74
	Expected	32.00	42.0	74.0

χ^2 Tests

	Value	df	p
χ^2	2.31	1	0.129
N	74		

Nominal

	Value
Phi-coefficient	0.177
Cramer's V	0.177

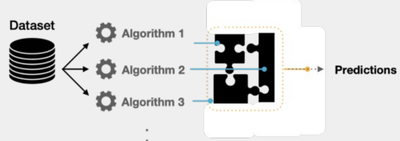
Independent Samples T-Test

		statistic	df	p	Cohen's d
AGE	Student's t	1.1306	72.0	0.262	0.2915
	Mann-Whitney U	458		0.238	0.2915
PTEDUCAT	Student's t	0.0515	72.0	0.959	0.0133
	Mann-Whitney U	530		0.752	0.0133

Group Descriptives

	Group	N	Mean	Median	SD	SE
AGE	FP_	21	74.4	74.6	7.38	1.610
	TN_	53	72.3	72.5	7.31	1.003
PTEDUCAT	FP_	21	16.3	16.0	3.38	0.737
	TN_	53	16.2	16.0	2.91	0.399

[X] Exactly the same as from Random Forest



Comparative statistics for correct vs incorrect classified sMCI (demographics)

“**True” stable MCI** who were **wrongfully classified as converting to AD** (TN vs FP) had**:**

- Worse score on all three memory tests
- Worse on TMT-B

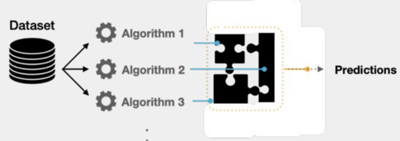
Group Descriptives

	Group	N	Mean	Median	SD	SE
RAVLT_immediate	FP_	21	31.00000	30.00000	7.28	1.589
	TN_	53	39.15094	39.00000	8.93	1.227
AVDEL30MIN_neuro	FP_	21	1.33333	1.00000	1.62	0.354
	TN_	53	5.92453	6.00000	3.44	0.472
AVDELTOT_neuro	FP_	21	9.47619	11.00000	3.36	0.732
	TN_	53	12.16981	13.00000	2.68	0.368
TRAASCOR_neuro	FP_	21	40.23810	42.00000	8.47	1.848
	TN_	53	37.32075	33.00000	13.19	1.812
TRABSCOR_neuro	FP_	21	128.80952	113.00000	59.87	13.065
	TN_	53	91.22642	82.00000	32.06	4.404
CATANIMSC_neuro	FP_	21	16.90476	16.00000	5.23	1.142
	TN_	53	18.67925	18.00000	4.87	0.669
LRHHC_n_long	FP_	21	0.00387	0.00404	6.20e−4	1.35e−4
	TN_	53	0.00457	0.00449	7.44e−4	1.02e−4
ANARTERR_neuro	FP_	21	9.09524	8.00000	7.13	1.555
	TN_	53	13.11321	10.00000	9.76	1.340
GDTOTAL_gds	FP_	21	1.90476	2.00000	1.14	0.248
	TN_	53	1.79245	2.00000	1.35	0.185

Independent Samples T-Test

		statistic	df	p	Cohen's d
RAVLT_immediate	Student's t	−3.716	72.0	<.001	−0.9581
	Mann-Whitney U	271		<.001	−0.9581
AVDEL30MIN_neuro	Student's t	−5.853 [a]	72.0	<.001	−1.5092
	Mann-Whitney U	114		<.001	−1.5092
AVDELTOT_neuro	Student's t	−3.623	72.0	<.001	−0.9341
	Mann-Whitney U	281		<.001	−0.9341
TRAASCOR_neuro	Student's t	0.937	72.0	0.352	0.2417
	Mann-Whitney U	413		0.086	0.2417
TRABSCOR_neuro	Student's t	3.496 [a]	72.0	<.001	0.9015
	Mann-Whitney U	323		0.005	0.9015
CATANIMSC_neuro	Student's t	−1.384	72.0	0.171	−0.3567
	Mann-Whitney U	447		0.188	−0.3567
LRHHC_n_long	Student's t	−3.820	72.0	<.001	−0.9851
	Mann-Whitney U	276		<.001	−0.9851
ANARTERR_neuro	Student's t	−1.712	72.0	0.091	−0.4414
	Mann-Whitney U	428		0.123	−0.4414
GDTOTAL_gds	Student's t	0.337	72.0	0.737	0.0868
	Mann-Whitney U	524		0.692	0.0868

[X] exactly the same group differences as from RF



Comparative statistics for correct vs incorrect classified sMCI (biological markers)

“True” stable MCI who were **wrongfully classified as converting to AD** (TN vs FP) had**:**

- Smaller hippocampus
- No difference in ApoE

Independent Samples T-Test

		statistic	df	p	Cohen's d
LRHHC_n_long	Student's t	−3.82	72.0	<.001	−0.985
	Mann-Whitney U	276		<.001	−0.985

Group Descriptives

	Group	N	Mean	Median	SD	SE
LRHHC_n_long	FP_	21	0.00387	0.00404	6.20e−4	1.35e−4
	TN_	53	0.00457	0.00449	7.44e−4	1.02e−4

Contingency Tables

		Apoe4_		
Ens_pred		0	1	Total
FP_	Observed	9	12	21
	Expected	12.2	8.80	21.0
TN_	Observed	34	19	53
	Expected	30.8	22.20	53.0
Total	Observed	43	31	74
	Expected	43.0	31.00	74.0

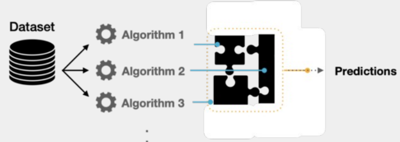
χ^2 Tests

	Value	df	p
χ^2	2.80	1	0.094
N	74		

Nominal

	Value
Phi-coefficient	0.195
Cramer's V	0.195

[X] Exactly the same as from Random Forest



Comparative statistics for correct vs incorrect classified sMCI (diagnostics)

“**True” stable MCI** who were **wrongfully classified as converting to AD** (TN vs FP) had**:**

- Scores worse only on Logical Memory

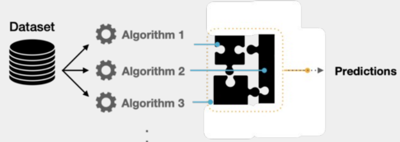
Independent Samples T-Test

		statistic	df	p
MMSE	Student's t	−1.9279	72.0	0.058
	Mann-Whitney U	401		0.057
CDRSB	Student's t	−0.0381	72.0	0.970
	Mann-Whitney U	491		0.416
LDELTOTAL	Student's t	−2.3745	72.0	0.020
	Mann-Whitney U	383		0.036
FAQTOTAL_faq	Student's t	0.5612	72.0	0.576
	Mann-Whitney U	458		0.217

Group Descriptives

	Group	N	Mean	Median	SD	SE
MMSE	FP_	21	27.24	28.00	1.841	0.402
	TN_	53	28.11	29.00	1.728	0.237
CDRSB	FP_	21	1.24	1.00	0.464	0.101
	TN_	53	1.25	1.00	0.812	0.112
LDELTOTAL	FP_	21	5.62	6.00	3.008	0.656
	TN_	53	7.23	8.00	2.462	0.338
FAQTOTAL_faq	FP_	21	2.76	2.00	3.740	0.816
	TN_	53	2.17	1.00	4.219	0.579

[ ]



Comparative statistics for correct vs incorrect classified cAD (demographics)

"**True" stable MCI** who were **wrongfully classified as converting to AD** (TN vs FP) had**:**

- No difference in any demographics

Independent Samples T-Test

		statistic	df	p	Cohen's d
AGE	Student's t	0.836	63.0	0.406	0.217
	Mann-Whitney U	438		0.537	0.217
PTEDUCAT	Student's t	1.012	63.0	0.315	0.263
	Mann-Whitney U	410		0.315	0.263

Group Descriptives

	Group	N	Mean	Median	SD	SE
AGE	FN_	23	75.0	73.6	8.15	1.700
	TP_	42	73.3	74.3	7.52	1.160
PTEDUCAT	FN_	23	16.2	17.0	2.79	0.582
	TP_	42	15.5	16.0	2.72	0.420

Contingency Tables

		PTGENDER		
Ens_pred		Female	Male	Total
FN_	Observed	7	16	23
	Expected	8.85	14.2	23.0
TP_	Observed	18	24	42
	Expected	16.15	25.8	42.0
Total	Observed	25	40	65
	Expected	25.00	40.0	65.0

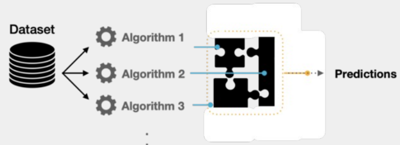
χ^2 Tests

	Value	df	p
χ^2	0.969	1	0.325
N	65		

Nominal

	Value
Phi-coefficient	0.122
Cramer's V	0.122

[X] Same as for the Random Frest



Comparative statistics for correct vs incorrect classified cAD (cog. feat. used in model)

“True” stable MCI who were **wrongfully classified as converting to AD** (TN vs FP) had**:**

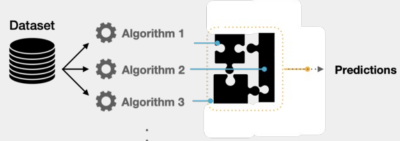
Group Descriptives

	Group	N	Mean	Median	SD	SE
RAVLT_immediate	FN_	23	35.22	34.00	7.73	1.611
	TP_	42	28.38	29.50	5.06	0.780
AVDEL30MIN_neuro	FN_	23	4.43	4.00	3.06	0.638
	TP_	42	1.29	1.00	1.55	0.239
AVDELTOT_neuro	FN_	23	11.61	12.00	2.61	0.544
	TP_	42	8.79	10.00	3.68	0.568
TRAASCOR_neuro	FN_	23	43.00	36.00	29.49	6.150
	TP_	42	44.86	37.00	25.50	3.934
TRABSCOR_neuro	FN_	23	122.65	78.00	82.64	17.231
	TP_	42	138.33	114.50	80.65	12.444
CATANIMSC_neuro	FN_	23	16.04	16.00	4.32	0.901
	TP_	42	15.79	16.00	4.19	0.646
ANARTERR_neuro	FN_	23	12.96	9.00	10.16	2.118
	TP_	42	13.29	9.00	10.16	1.568
GDTOTAL_gds	FN_	23	1.30	1.00	1.15	0.239
	TP_	42	1.50	1.00	1.23	0.191

Independent Samples T-Test

		statistic	df	p	Cohen's d
RAVLT_immediate	Student's t	4.304 [a]	63.0	<.001	1.1165
	Mann-Whitney U	230		<.001	1.1165
AVDEL30MIN_neuro	Student's t	5.524 [a]	63.0	<.001	1.4330
	Mann-Whitney U	178		<.001	1.4330
AVDELTOT_neuro	Student's t	3.254 [a]	63.0	0.002	0.8442
	Mann-Whitney U	262		0.002	0.8442
TRAASCOR_neuro	Student's t	−0.266	63.0	0.791	−0.0689
	Mann-Whitney U	401		0.260	−0.0689
TRABSCOR_neuro	Student's t	−0.743	63.0	0.460	−0.1928
	Mann-Whitney U	389		0.199	−0.1928
CATANIMSC_neuro	Student's t	0.235	63.0	0.815	0.0609
	Mann-Whitney U	482		0.989	0.0609
ANARTERR_neuro	Student's t	−0.125	63.0	0.901	−0.0324
	Mann-Whitney U	471		0.874	−0.0324
GDTOTAL_gds	Student's t	−0.626	63.0	0.533	−0.1625
	Mann-Whitney U	441		0.551	−0.1625

[X] Same pattern as from RF



Comparative statistics for correct vs incorrect classified cAD (biological markers)

“**True” stable MCI** who were **wrongfully classified as converting to AD** (TN vs FP) had**:**

- Smaller hippocampus
- Lower proportion of ApoE negative subjects

Contingency Tables

Ens_pred		Apoe4_ 0	1	Total
FN_	Observed	13	10	23
	Expected	7.78	15.2	23.0
TP_	Observed	9	33	42
	Expected	14.22	27.8	42.0
Total	Observed	22	43	65
	Expected	22.00	43.0	65.0

χ^2 Tests

	Value	df	p
χ^2	8.17	1	0.004
N	65		

Nominal

	Value
Phi-coefficient	0.355
Cramer's V	0.355

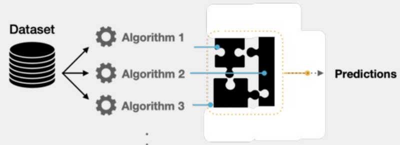
Independent Samples T-Test

		statistic	df	p	Cohen's d
LRHHC_n_long	Student's t	3.72	63.0	<.001	0.965
	Mann-Whitney U	218		<.001	0.965

Group Descriptives

	Group	N	Mean	Median	SD	SE
LRHHC_n_long	FN_	23	0.00439	0.00427	6.37e–4	1.33e–4
	TP_	42	0.00375	0.00366	6.81e–4	1.05e–4

[O] Not the same as from RF



Comparative statistics for correct vs incorrect classified cAD (diagnostics)

“**True” stable MCI** who were **wrongfully classified as converting to AD** (TN vs FP) had**:**

- Worse scores on all except MMSE
  - Again, opposite pattern from sMCI

Independent Samples T-Test

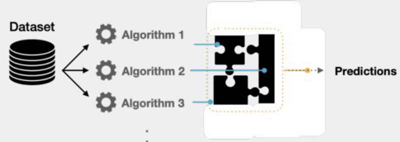
		statistic	df	p	Cohen's d
MMSE	Student's t	1.61	63.0	0.112	0.419
	Mann-Whitney U	373		0.125	0.419
CDRSB	Student's t	–2.49	63.0	0.016	–0.645
	Mann-Whitney U	317		0.020	–0.645
LDELTOTAL	Student's t	4.41 [a]	63.0	< .001	1.143
	Mann-Whitney U	221		< .001	1.143
FAQTOTAL_faq	Student's t	–2.33 [a]	63.0	0.023	–0.604
	Mann-Whitney U	310		0.017	–0.604

[a] Levene's test is significant ($p < .05$), suggesting a violation of the assumption of equal variances

Group Descriptives

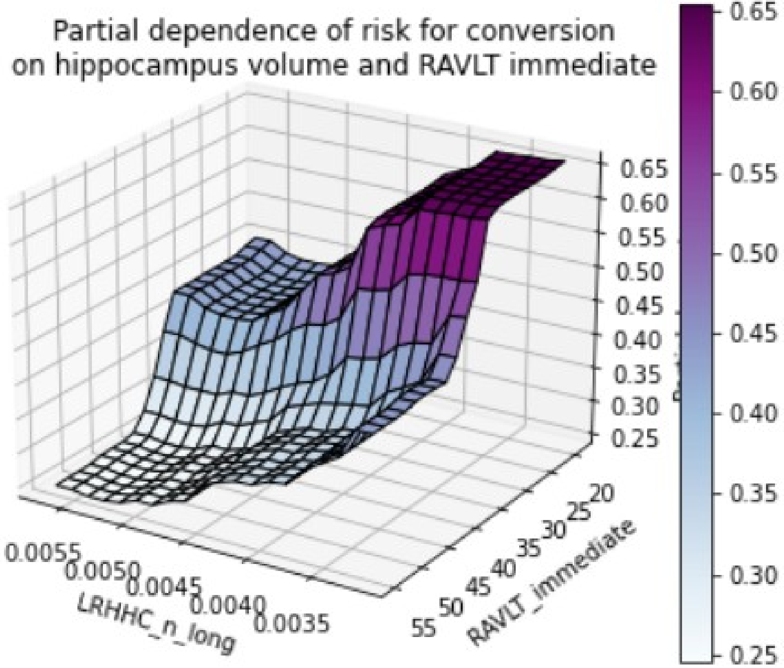
	Group	N	Mean	Median	SD	SE
MMSE	FN_	23	27.30	28.00	1.917	0.400
	TP_	42	26.57	27.00	1.655	0.255
CDRSB	FN_	23	1.59	1.50	0.701	0.146
	TP_	42	2.13	2.00	0.911	0.141
LDELTOTAL	FN_	23	6.00	6.00	3.464	0.722
	TP_	42	2.76	2.00	2.428	0.375
FAQTOTAL_faq	FN_	23	2.65	2.00	3.142	0.655
	TP_	42	5.33	4.00	4.996	0.771

[X] Same as from RF

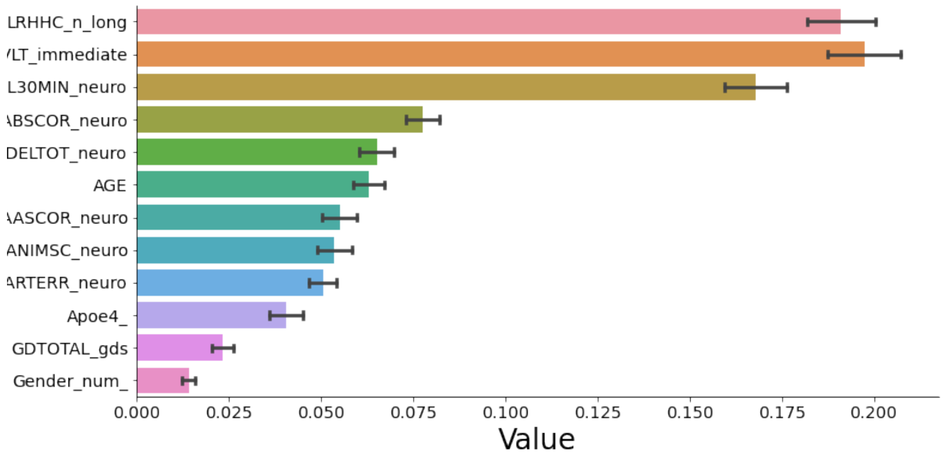


Looking inside the **Random Forest** “Black Box” on th

Får plutselig ikke til å lage disse plottene lengre, lurer på om det kan ha noe å gjøre med versjoner av packages som muligens ble endret når jeg innstralerte PyCaret...



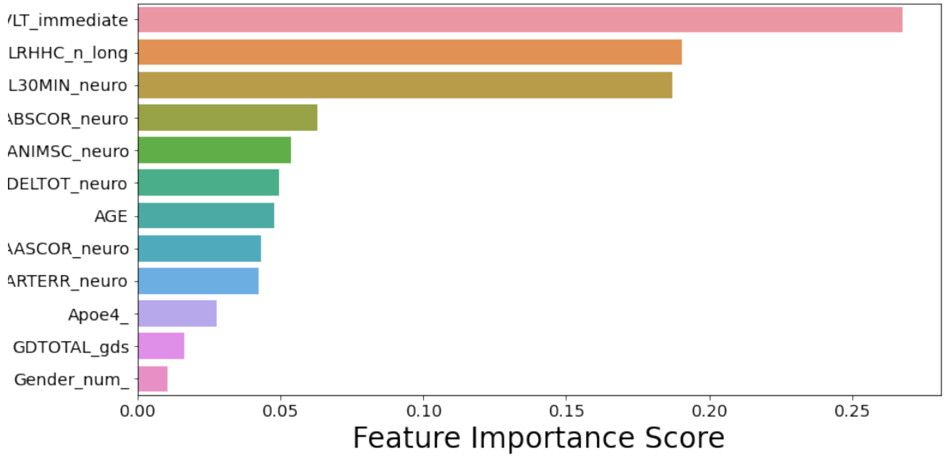
Mean feature importance for k=50 folds



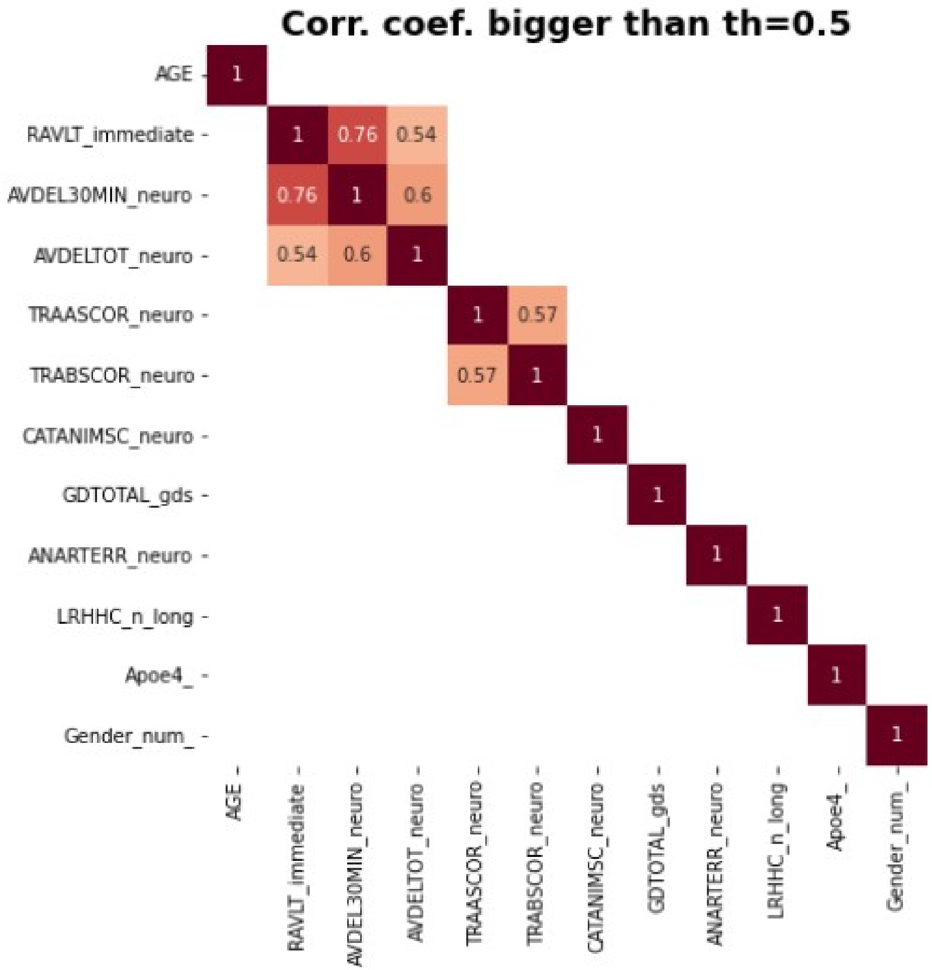
[65]:

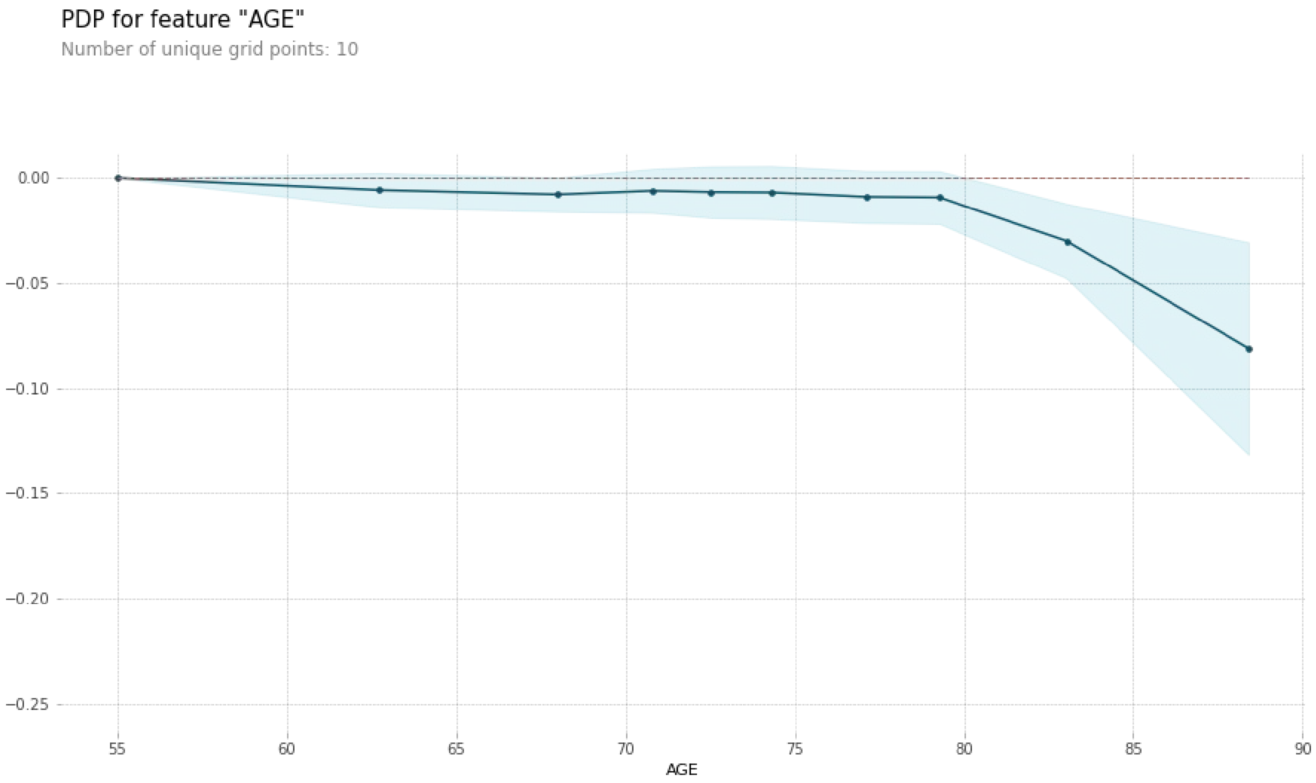
Weight	Feature
0.0403 ± 0.0503	LRHHC_n_long
0.0245 ± 0.0413	RAVLT_immediate
0.0086 ± 0.0108	AGE
0.0058 ± 0.0141	CATANIMSC_neuro
0.0000 ± 0.0129	Gender_num_
-0.0014 ± 0.0211	Apoe4_
-0.0014 ± 0.0058	GDTOTAL_gds
-0.0029 ± 0.0503	AVDEL30MIN_neuro
-0.0058 ± 0.0168	TRAASCOR_neuro
-0.0101 ± 0.0147	TRABSCOR_neuro
-0.0129 ± 0.0058	ANARTERR_neuro
-0.0158 ± 0.0279	AVDELTOT_neuro

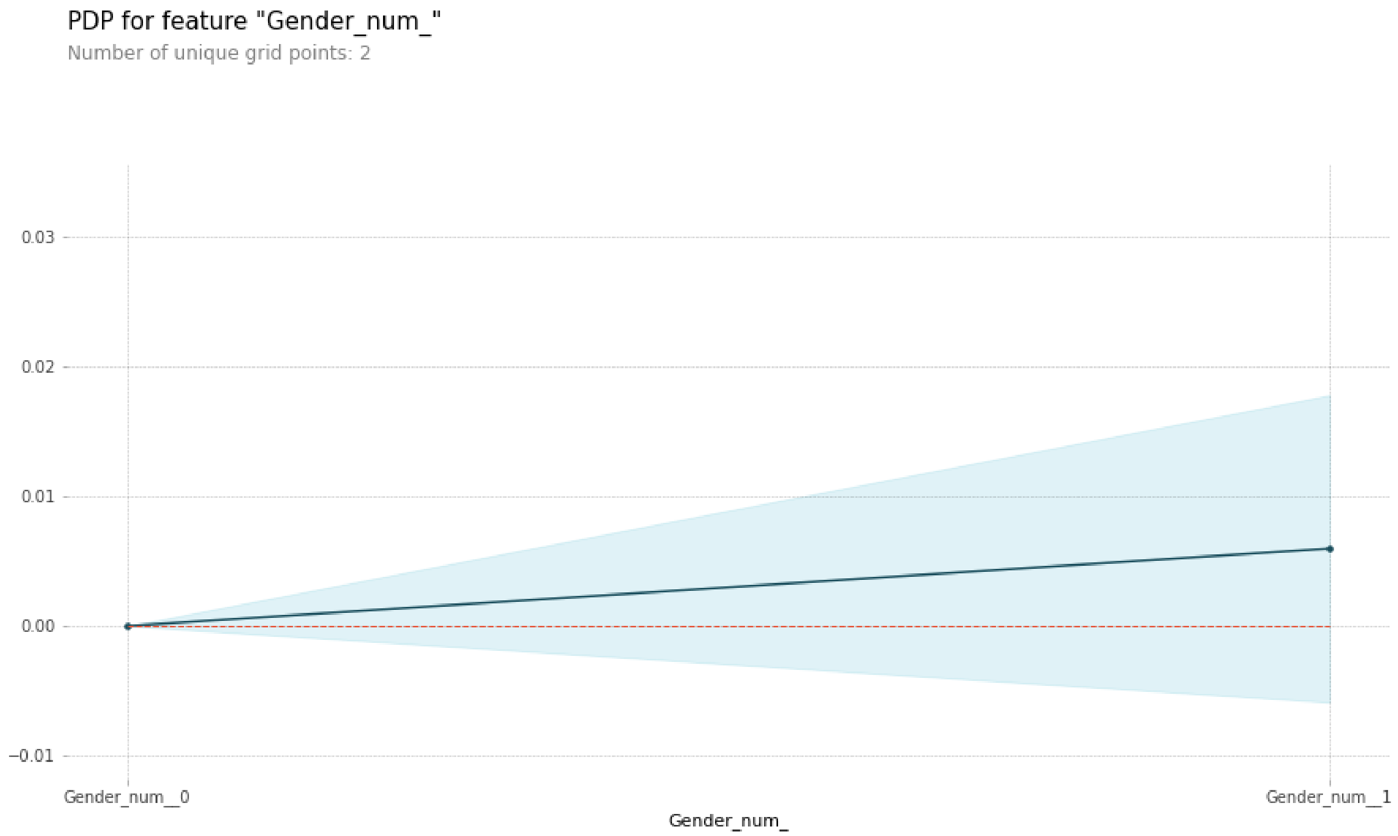
RF Feature Importance

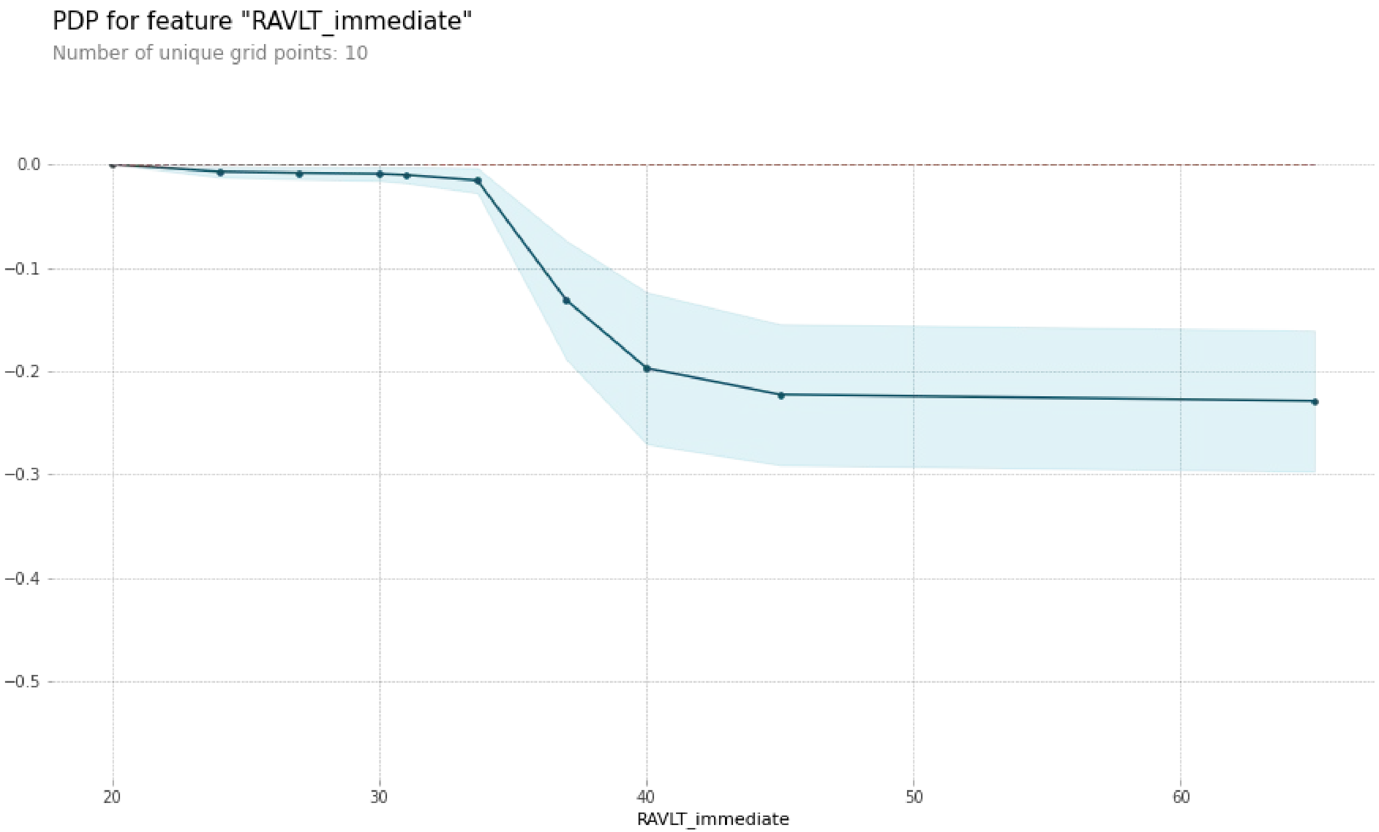


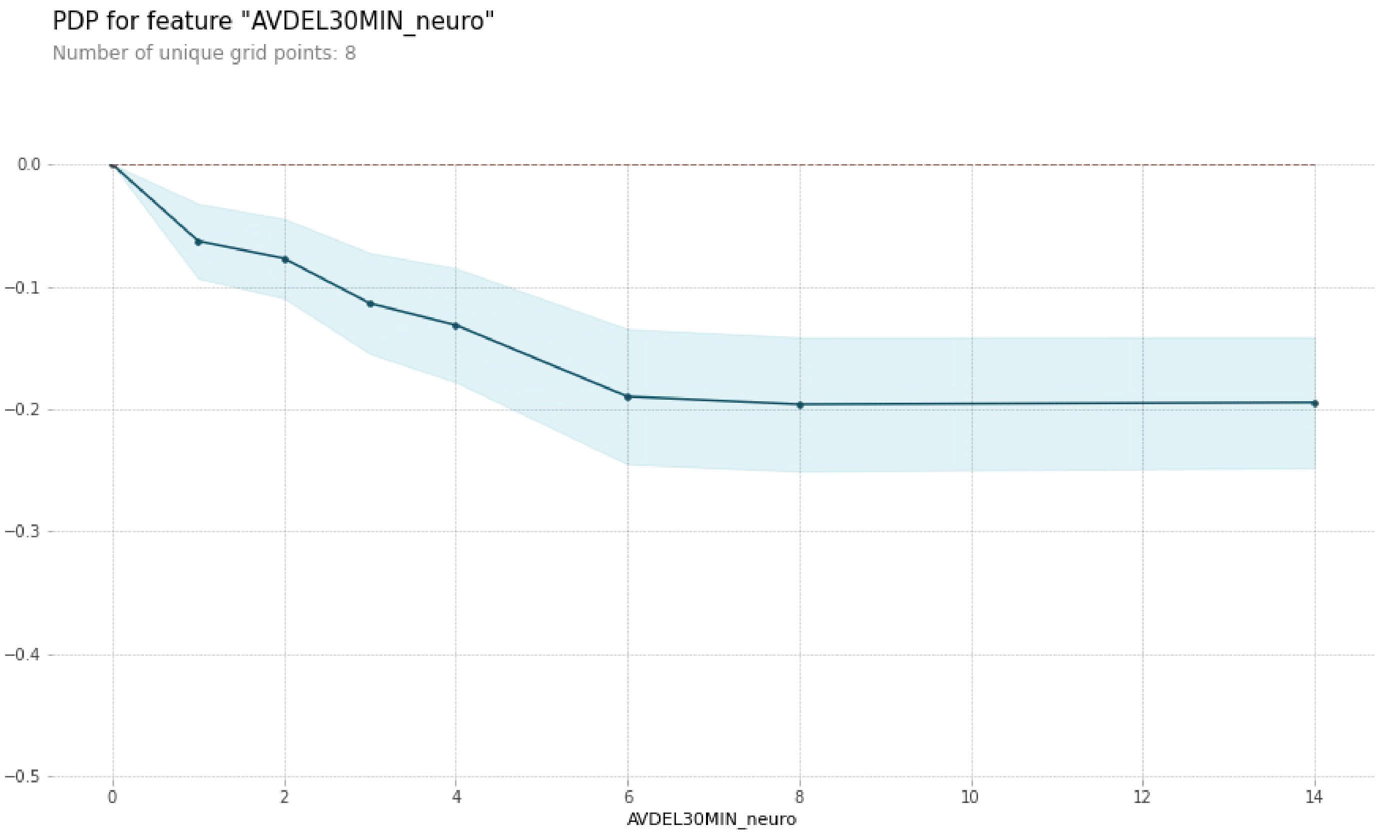
← From TEST set

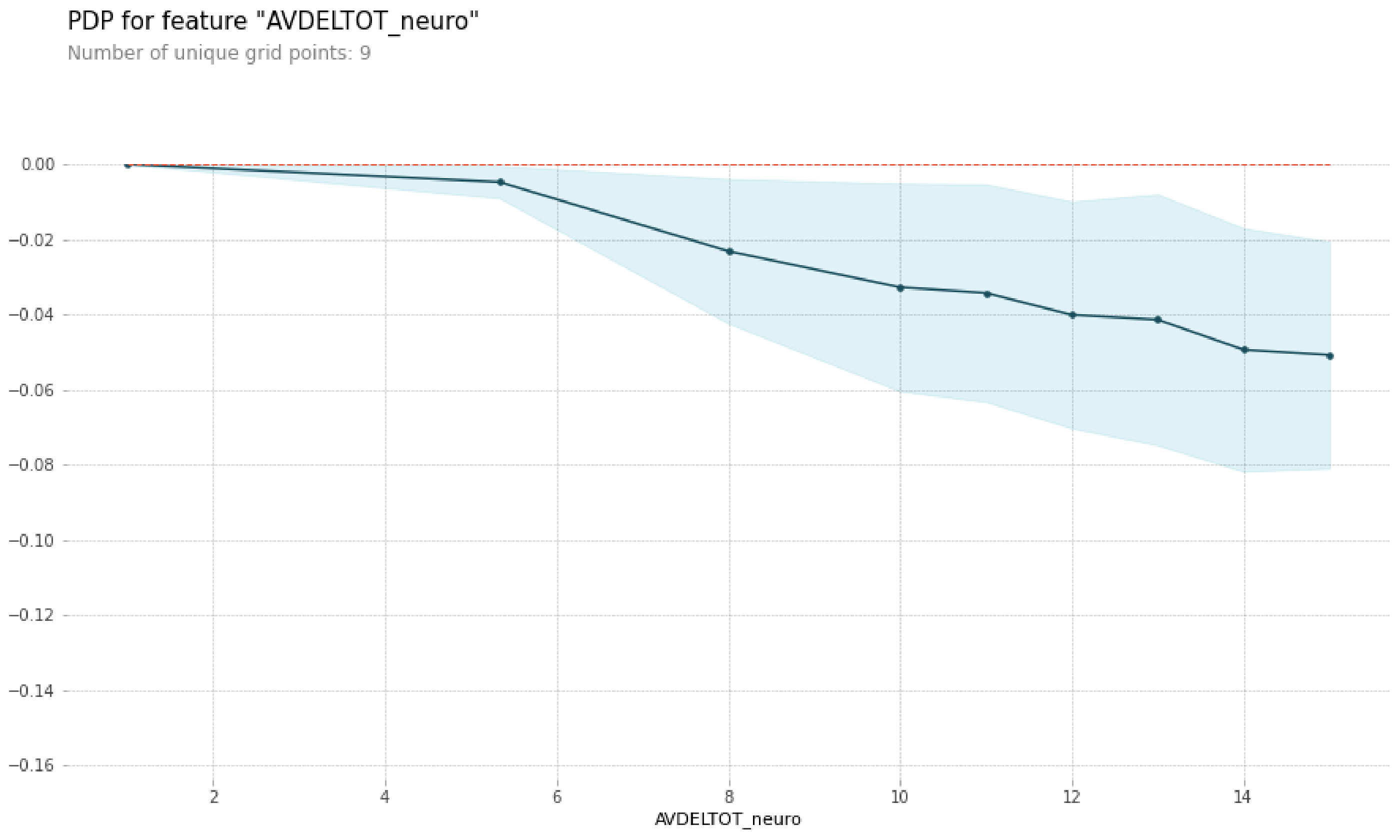

Corr. coef. bigger than th=0.5
AGE
RAVLT_immediate
AVDEL30MIN_neuro
AVDELTOT_neuro
TRAASCOR_neuro
TRABSCOR_neuro
CATANIMSC_neuro
GDTOTAL_gds
ANARTERR_neuro
LRHHC_n_long
Apoe4_
Gender_num_
1
0.76
0.54
0.6
0.57

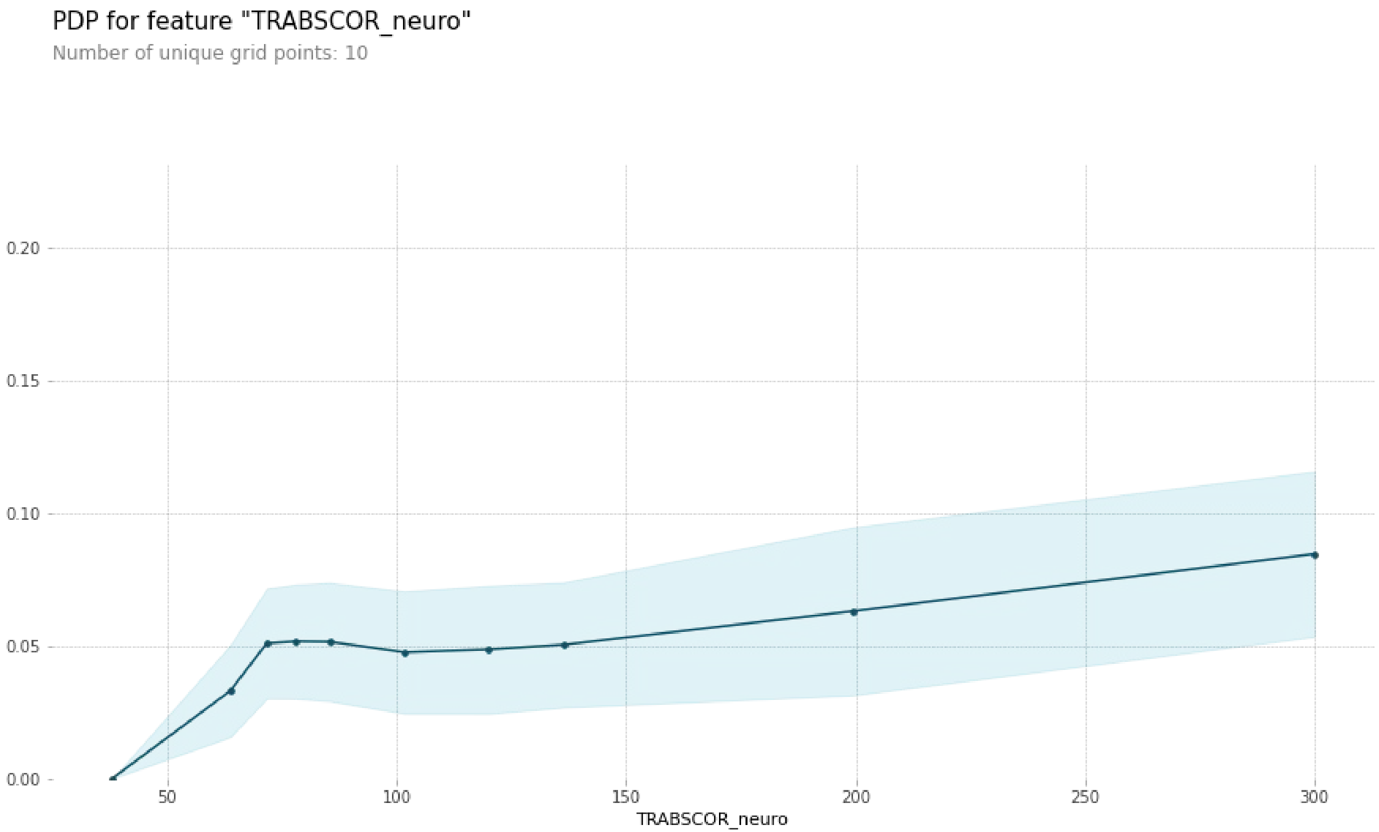

PDP for feature "AGE"
Number of unique grid points: 10
0.00
−0.05
−0.10
−0.15
−0.20
−0.25
55
60
65
70
75
80
85
90
AGE

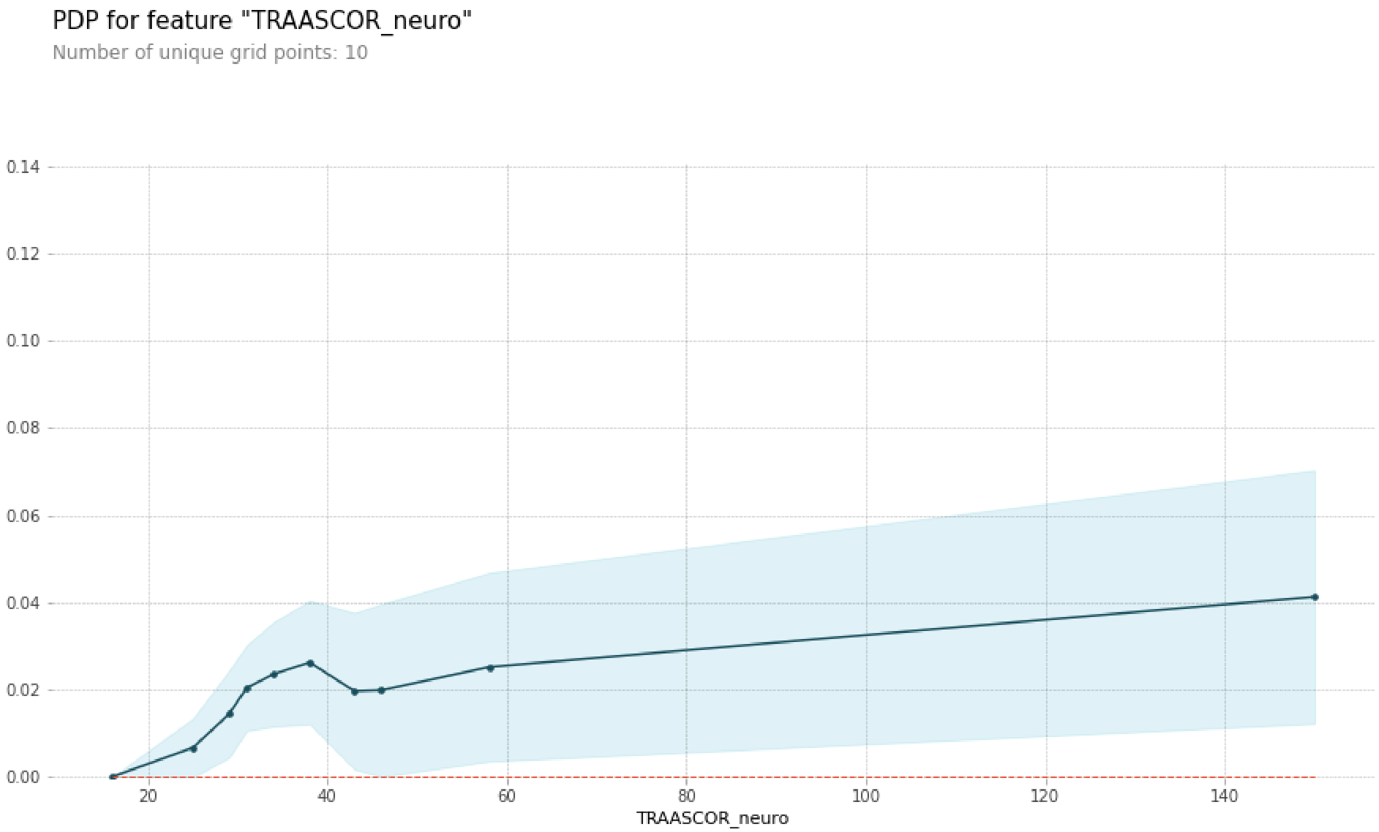

PDP for feature "Gender_num_"
Number of unique grid points: 2
0.03
0.02
0.01
0.00
−0.01
Gender_num__0
Gender_num__1
Gender_num_

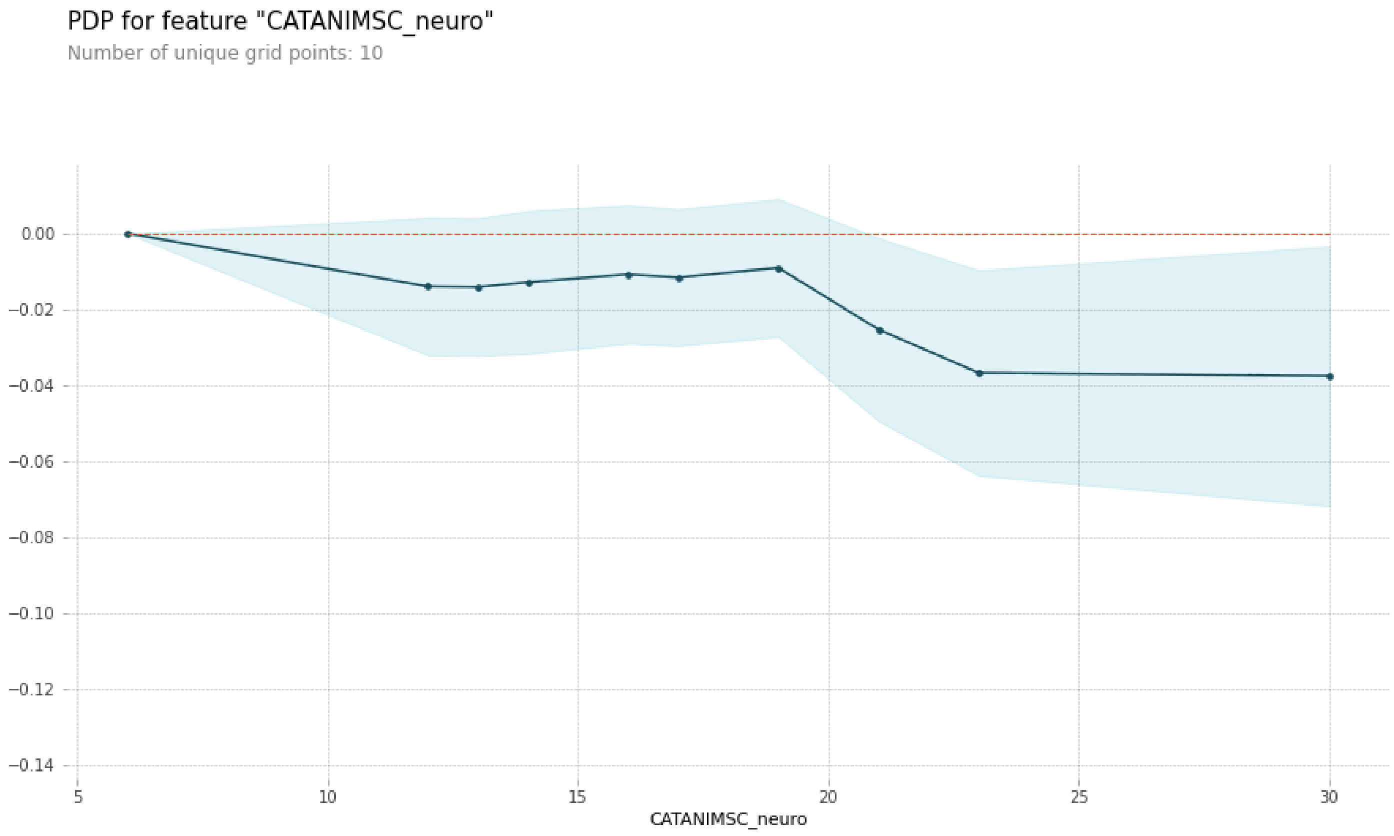

PDP for feature "RAVLT_immediate"
Number of unique grid points: 10
0.0
−0.1
−0.2
−0.3
−0.4
−0.5
20
30
40
50
60
RAVLT_immediate

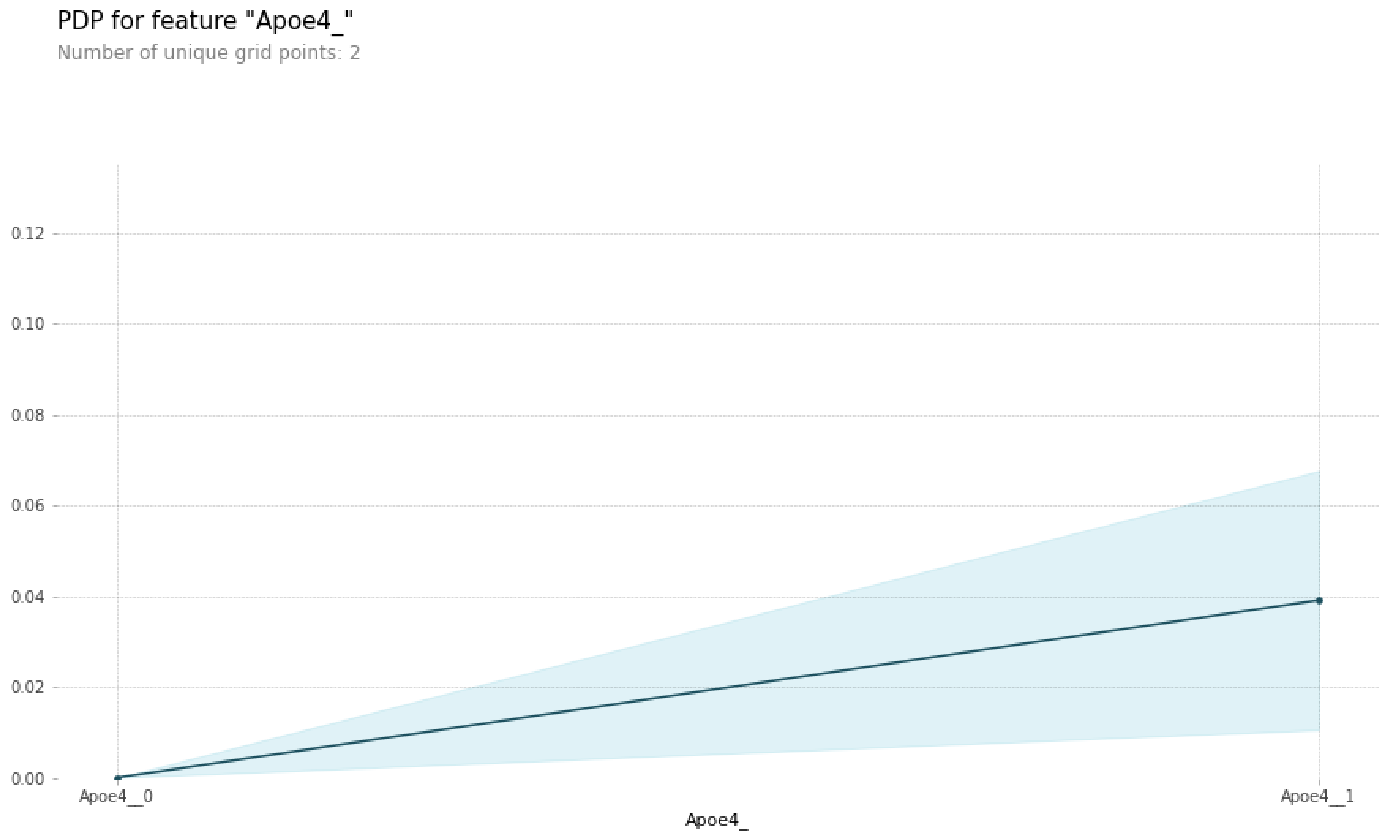

PDP for feature "AVDEL30MIN_neuro"
Number of unique grid points: 8
0.0
−0.1
−0.2
−0.3
−0.4
−0.5
0
2
4
6
8
10
12
14
AVDEL30MIN_neuro

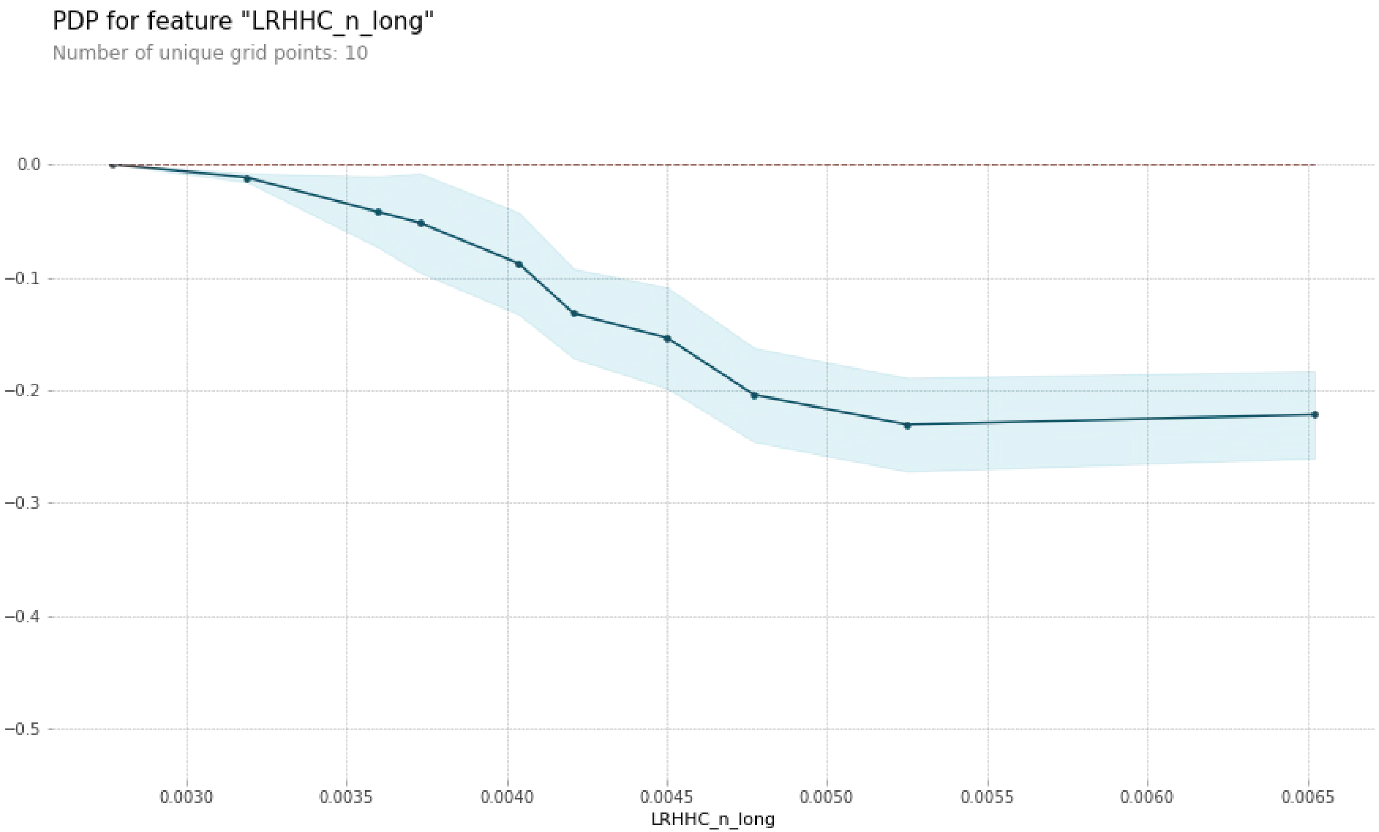

PDP for feature "AVDELTOT_neuro"
Number of unique grid points: 9
0.00
−0.02
−0.04
−0.06
−0.08
−0.10
−0.12
−0.14
−0.16
2
4
6
8
10
12
14
AVDELTOT_neuro


PDP for feature "TRABSCOR_neuro"
Number of unique grid points: 10
0.20
0.15
0.10
0.05
0.00
50
100
150
200
250
300
TRABSCOR_neuro

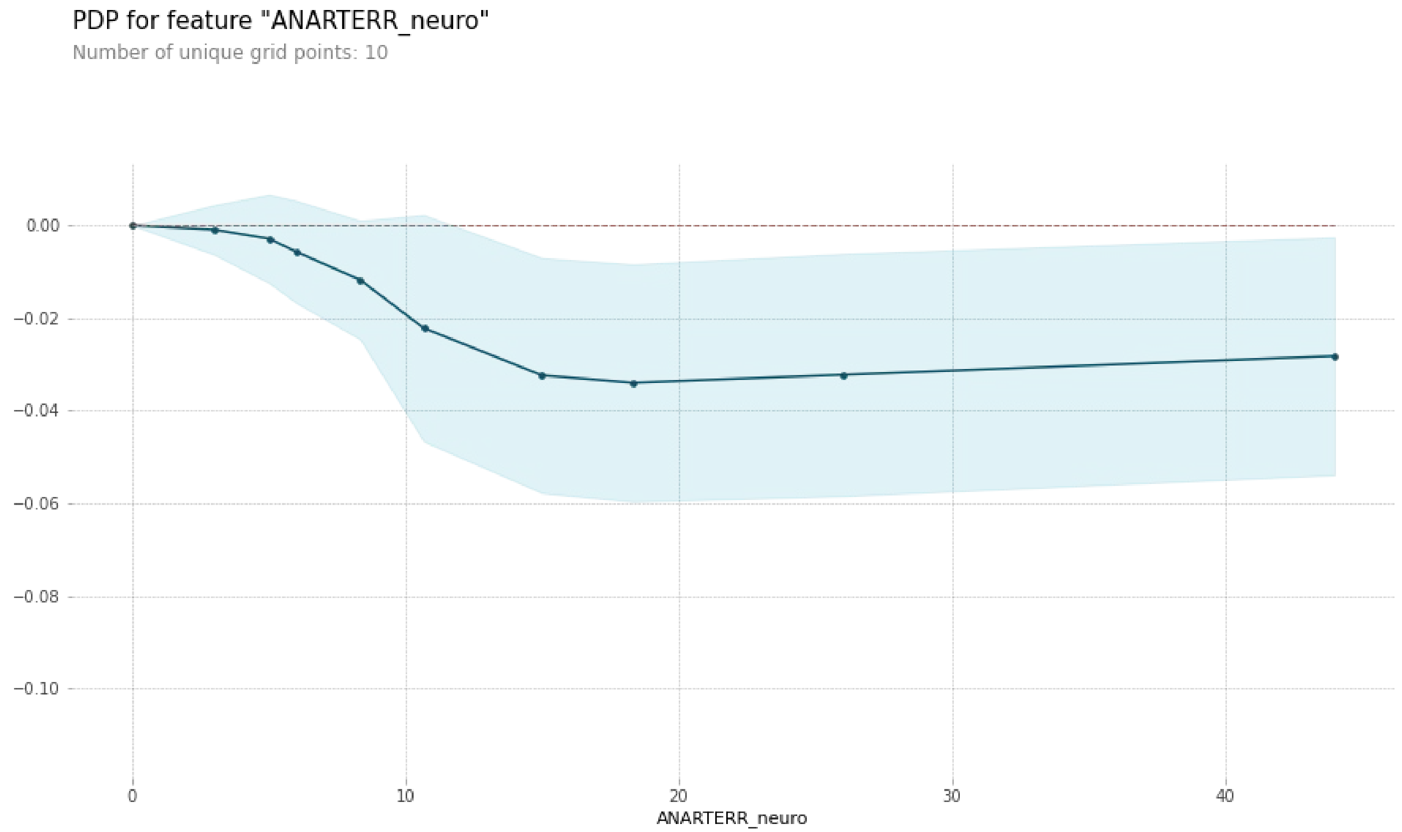

PDP for feature "TRAASCOR_neuro"
Number of unique grid points: 10
0.14
0.12
0.10
0.08
0.06
0.04
0.02
0.00
20
40
60
80
100
120
140
TRAASCOR_neuro


PDP for feature "CATANIMSC_neuro"
Number of unique grid points: 10
0.00
−0.02
−0.04
−0.06
−0.08
−0.10
−0.12
−0.14
5
10
15
20
25
30
CATANIMSC_neuro

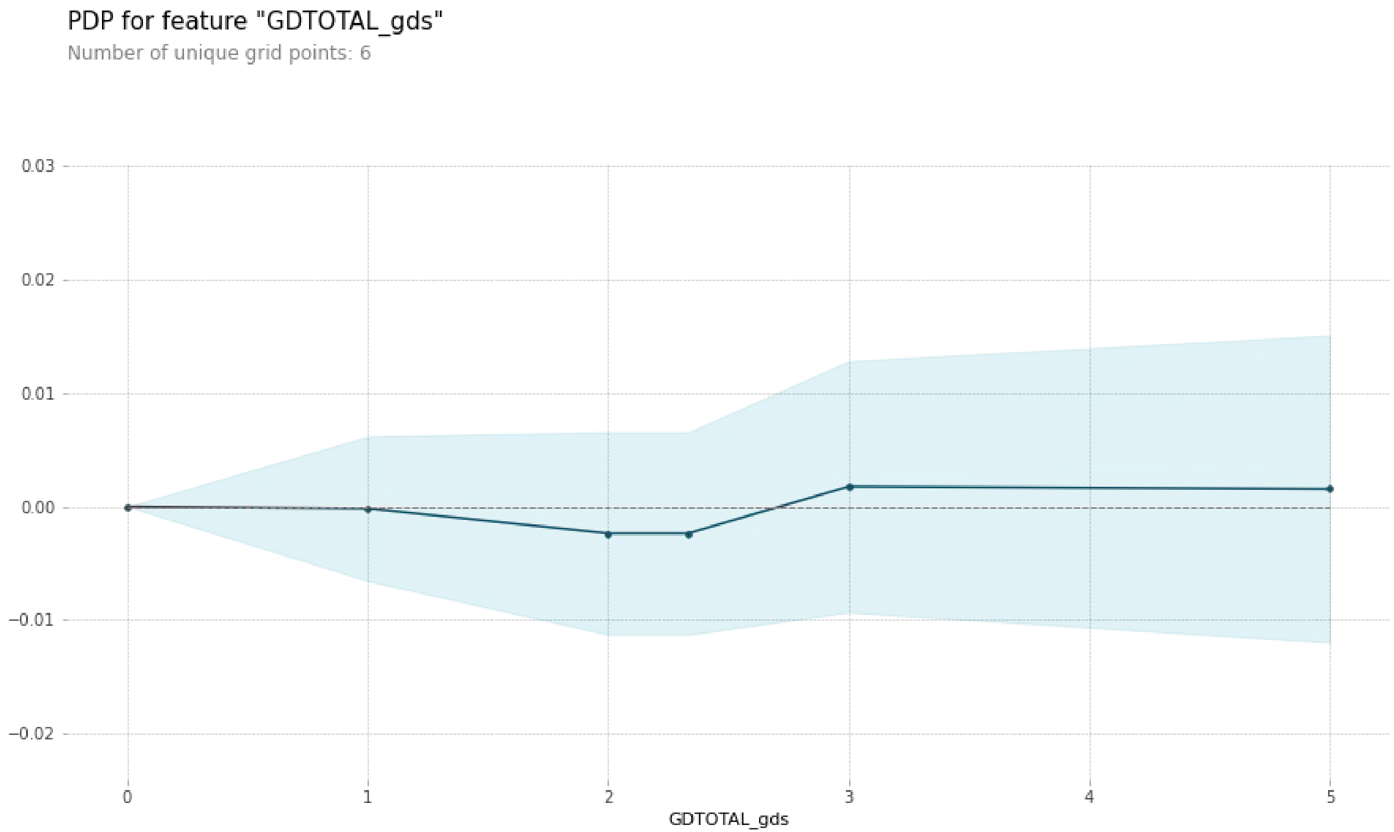

PDP for feature "Apoe4_"
Number of unique grid points: 2
0.12
0.10
0.08
0.06
0.04
0.02
0.00
Apoe4__0
Apoe4__1
Apoe4_



PDP for feature "LRHHC_n_long"
Number of unique grid points: 10
0.0
−0.1
−0.2
−0.3
−0.4
−0.5
0.0030
0.0035
0.0040
0.0045
0.0050
0.0055
0.0060
0.0065
LRHHC_n_long



PDP for feature "ANARTERR_neuro"
Number of unique grid points: 10
0.00
−0.02
−0.04
−0.06
−0.08
−0.10
0
10
20
30
40
ANARTERR_neuro


PDP for feature "GDTOTAL_gds"
Number of unique grid points: 6
0.03
0.02
0.01
0.00
−0.01
−0.02
0
1
2
3
4
5
GDTOTAL_gds