result = [permutation\_importance](https://scikit-learn.org/stable/modules/generated/sklearn.inspection.permutation_importance.html#sklearn.inspection.permutation_importance)(clf, X\_train, y\_train, n\_repeats=10, random\_state=42)

perm\_sorted\_idx = result.importances\_mean.argsort()

tree\_importance\_sorted\_idx = [np.argsort](https://numpy.org/doc/stable/reference/generated/numpy.argsort.html#numpy.argsort)(clf.feature\_importances\_)

tree\_indices = [np.arange](https://numpy.org/doc/stable/reference/generated/numpy.arange.html#numpy.arange)(0, len(clf.feature\_importances\_)) + 0.5

fig, (ax1, ax2) = [plt.subplots](https://matplotlib.org/api/_as_gen/matplotlib.pyplot.subplots.html#matplotlib.pyplot.subplots)(1, 2, figsize=(12, 8))

ax1.barh(tree\_indices, clf.feature\_importances\_[tree\_importance\_sorted\_idx], height=0.7)

ax1.set\_yticks(tree\_indices)

ax1.set\_yticklabels(data.feature\_names[tree\_importance\_sorted\_idx])

ax1.set\_ylim((0, len(clf.feature\_importances\_)))

ax2.boxplot(

result.importances[perm\_sorted\_idx].T,

vert=**False**,

labels=data.feature\_names[perm\_sorted\_idx],

)

fig.tight\_layout()

[plt.show](https://matplotlib.org/api/_as_gen/matplotlib.pyplot.show.html#matplotlib.pyplot.show)()

