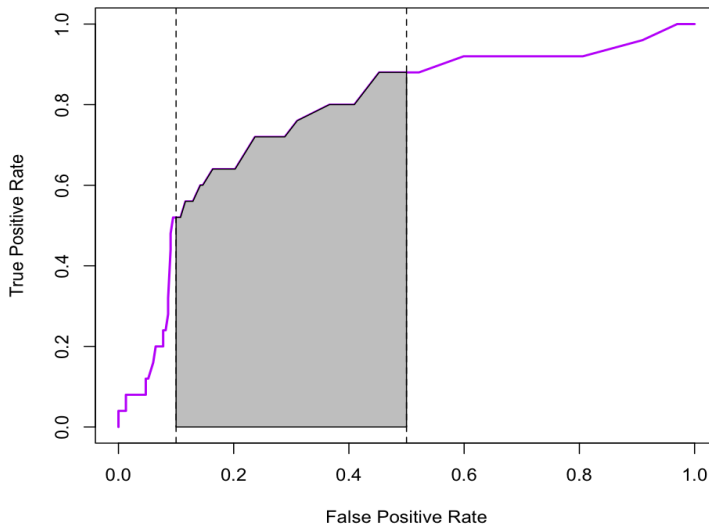


## Advanced R

### Exercises for Session 4: Objects and Methods.

1. (a) Write a `summary()` method for the S3 ROC curve object that calculates the *partial area under the curve* (pAUC). That is, given an upper and lower endpoint for the false positive rate (x-axis), it computes the area under the curve between the upper and lower endpoints. The endpoints should default to 0 and 1, and the user should be able to specify both, just one, or neither.



*Shaded area is partial AUC between false positive rates of 0.1 and 0.5*

(b) Repeat for the S4 ROC curve objects. For debugging, try using `trace()` with the `signature=` argument set to follow the S4 methods.

2. ROC curves are also useful for summarizing logistic regression models. Given a logistic regression model object `model` of class `glm`, we can extract the predicted values with `fitted(model)` and the observed binary outcome with `model$y` and then use the same computations to produce an ROC curve for the model.

(a) Write a function that computes the ROC curve for a logistic regression model.

(b) Write a *generic* function `ROC()` with methods that compute the ROC curve when the user supplies *either* a continuous test variable and binary outcome *or* a `glm` object.

3. The data file `sequence.rda` has DNA sequences for 5000 people and 4028 *rare* sequence variants (simulated using MaCS). The Matrix package provides tools for handling sparse (ie, mostly zero) matrices. Compare the time and memory taken for

```
crossprod(sequence)
```

and

```
library(Matrix)
M<-Matrix(sequence)
crossprod(M)
```

Look at the class of `M` and the methods available for that class. What can you find out about how `M` is actually stored internally?