Prediction & Validation

Overview

- Verification/validation/classification

- Predictions and Interpretations

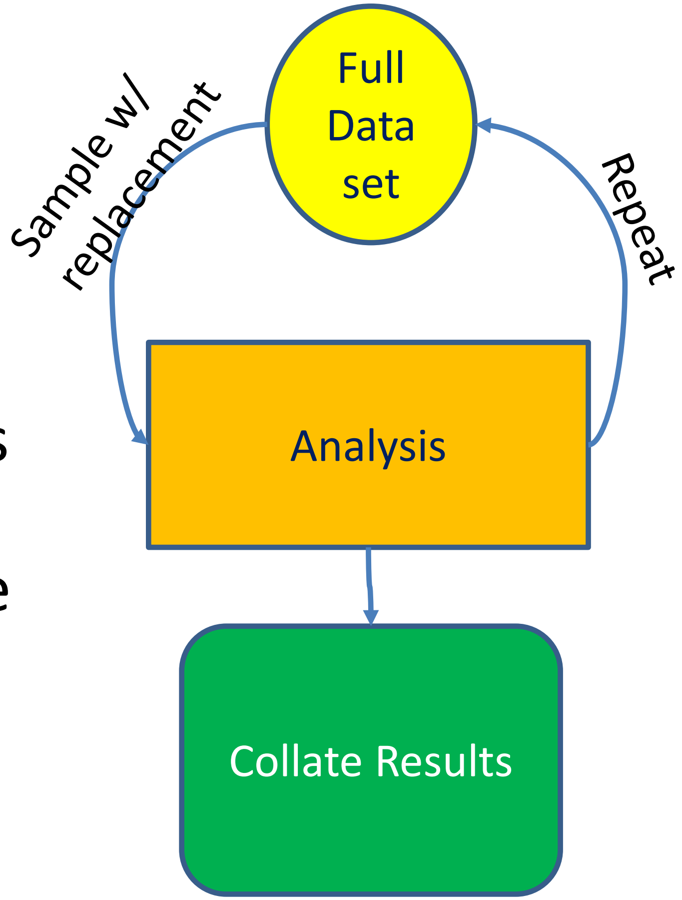
Verification

- Verification – ‘the process of establishing the truth, accuracy, or validity of something.’
- For this model methodology: we look at getting a more thorough understanding of the model.

Bootstrapping [1]

- Widely used methodology for understanding parameters and uncertainty.

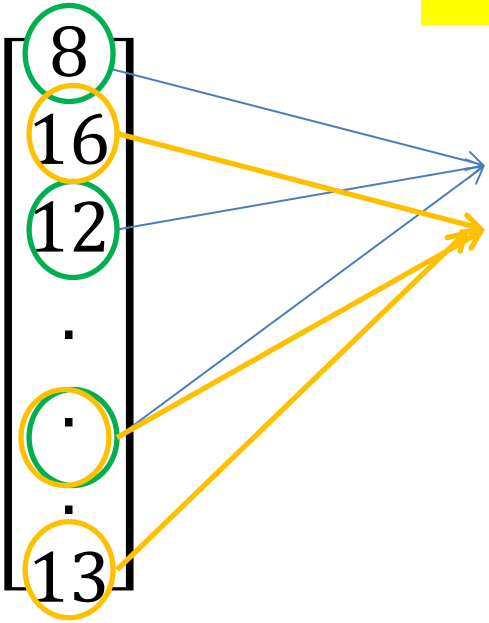
- General Methodology:
  - Random sampling of observations (with replacement).
  - Apply model to data and calculate parameters.
  - Repeat n times (usually n=1000)
  - Review distribution of parameter space.



Bootstrapping [2]

- Example – calculate median value of vector

Calculated $\text{median}_{\text{all}}$ (all of the data) = 24



Median_1 = 25
Median_2 = 24

Median_n = 22

Take quantiles

5% = 19
50% = 23.5
95% = 25

Bootstrapping [3]

How:

- Application in NONMEM via PsN etc.
- R packages: Boot
- Easy to construct a custom script for these models. (e.g.dataset[sample(nrow(dataset)**,replace=T),**])

```
# quick loop
for(i in 1:nBS) {
  bsData <- full[sample(nrow(full),replace=T),]
  bsMod <- clm(DV ~ EXP + SEX, data=bsData)
  coef.out[i,] <- coef(bsMod)
  llkhood.out[i] <- logLik(bsMod)
  aic.out[i] <- AIC(bsMod)
}
```

Remember to set your seed for reproducibility!!!

Bootstrapping [4]

Example:

- Parameter output

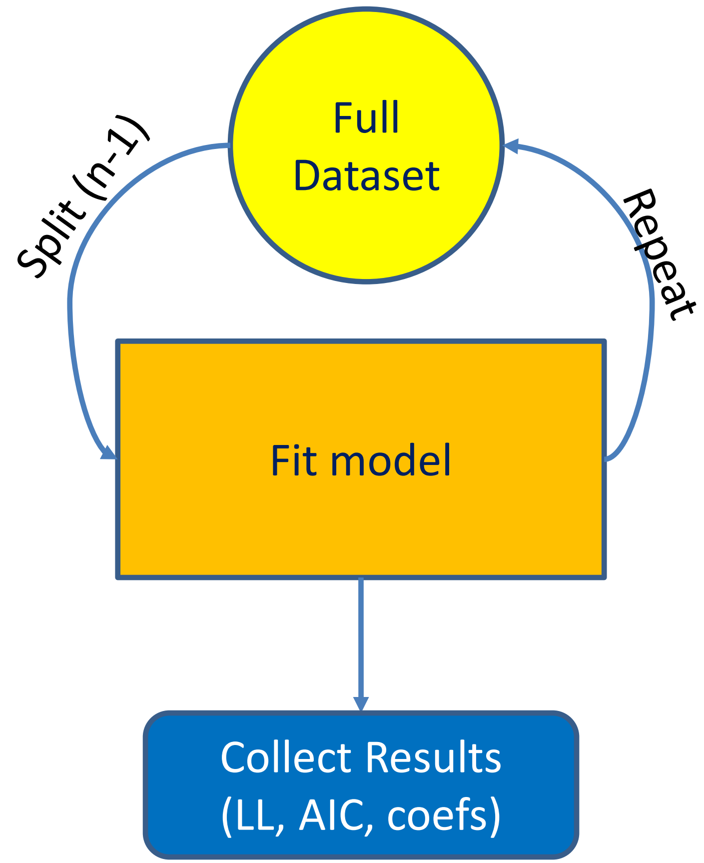
		Bootstrap Output		
Parameter	**Model**	**Median**	**Lower 5%**	**Upper 95%**
0|1	-1.95	-1.96	-2.57	-1.35
1|2	-0.338	-0.336	-0.929	0.203
2|3	1.66	1.67	1.10	2.28
3|4	2.35	2.37	1.74	3.04
EXP	0.0541	0.0545	0.035	0.077
SEX2	-1.04	-1.07	-1.63	-0.492
LL	-273	-270	-284	-255
AIC	557	553	522	580

- Model output and median are highly similar.
- Interval includes 0… (is this a concern for a cut point? – No)
- Range provides understanding on overall uncertainty.

Verification of model (or identify potential concerns)

Jackknife (CDD)

A Jack-knife or Case deletion diagnostic can be useful in understanding influential individuals.



General methodology:

- Split data into groups removing 1 individual for each group.
- Fit model
- Compare LL, AIC, … for each of outputs.

Jack… [2]

Like other diagnostics there are R packages, but again easy to customize…

Dataset with ID

Prep for output

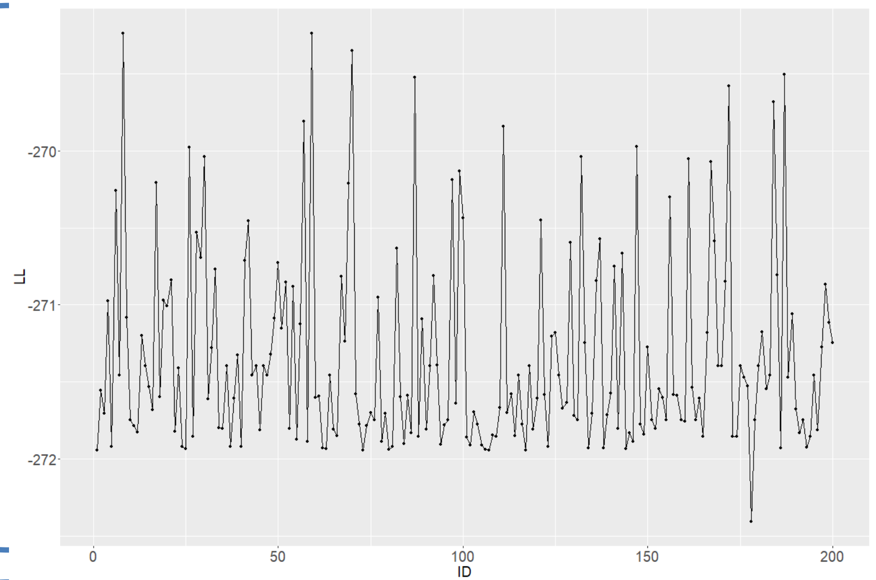
Collect output

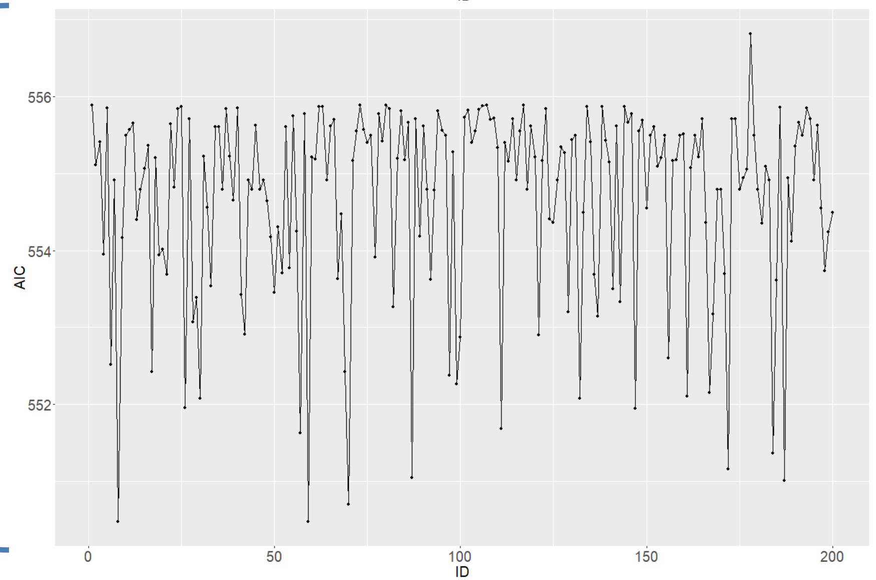
```
# use a dataset with ID
fullDat <- full
fullDat$ID <- 1:nrow(fullDat)

# input needed
nID <- nrow(fullDat)  # number of individuals


# prepare data collection
coef.out <- matrix(nrow=nID, ncol=length(coef(fit.final)))
llkhood.aic.out <- data.frame(ID=fullDat$ID,
                              LL=rep(NA, nrow(fullDat)),
                              aic=rep(NA, nrow(fullDat)))


# quick loop
for(i in 1:nID) {
  jkData <- fullDat[-i,]  #drop an individual each time
  jkMod <- clm(DV ~ EXP + SEX, data=jkData)
  coef.out[i,] <- coef(jkMod)
  llkhood.aic.out[i,2] <- logLik(jkMod)
  llkhood.aic.out[i,3]  <- AIC(jkMod)
}
```

Jack… [3]

<u>Review LL or AIC by ID</u>

- What is range?
- Specific ID with large anomalies?





- Looks Ok…

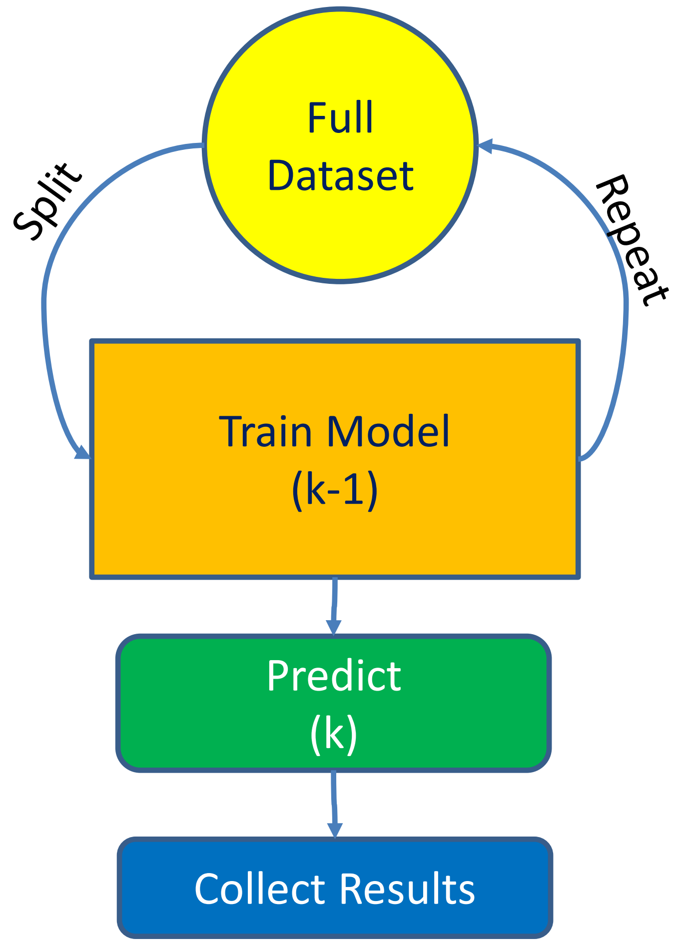
Cross Validation [1]

- Cross-validation methodology (k-fold, v-fold,…) proposed in a learn ➔ confirm paradigm.
- Cross-Validation is commonly used in machine learning, data mining, and other statistical approaches.
- Allows a dataset to be used for both model building and model validation.

Cross Validation [2]

General Methodology:

- Split the data into k equal components.
- Use k-1 of the components to train the model
- Use the remainder (k) to test the prediction.



Cross Validation [3]

Implementation:

- Again there are R packages (e.g. cvTools, caret)
- Easy to construct a custom script:
  - Split train and test data, model, collect…

```
for(i in 1:nCross){
  testIndexes <- which(folds==i,arr.ind = T)
  testData <- df[testIndexes,]
  names(testData)[8] <- "obsDV"
  trainData <- df[-testIndexes,]
  fin.model  <- clm(DV ~ EXP + SEX, data=trainData)
```

Cross Validation [4]

<u>Output</u>:

- Model parameters from each training model.
  - Need to review in case of influential values
- Predicted value (or probability of value) for each matched observation.
- For classification review output using [will discuss later]
  - ROC
  - ‘Confusion’ matrix and tables.

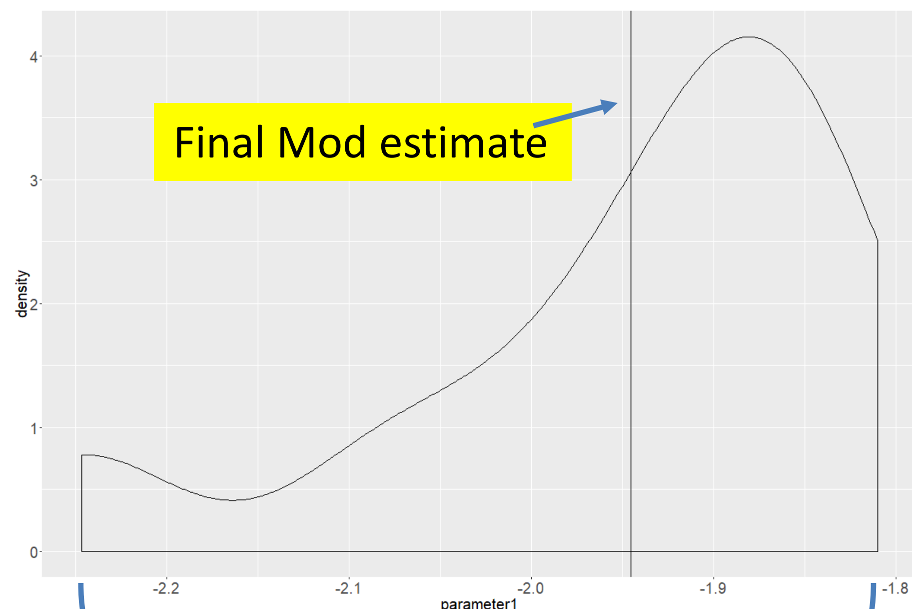
Cross Validation [5]

<u>Output</u>:

- Check model parameters.

- Are parameters consistent with understanding of model? -Yes

	Final	mean	SD
0|1	-1.94550345	-1.94788915	0.130954758
1|2	-0.33831495	-0.33835643	0.108844222
2|3	1.66112946	1.66359596	0.108877276
3|4	2.35256702	2.35633409	0.111379112
EXP	0.05410534	0.05425945	0.004140862
SEX2	-1.03798707	-1.04065229	0.088106040



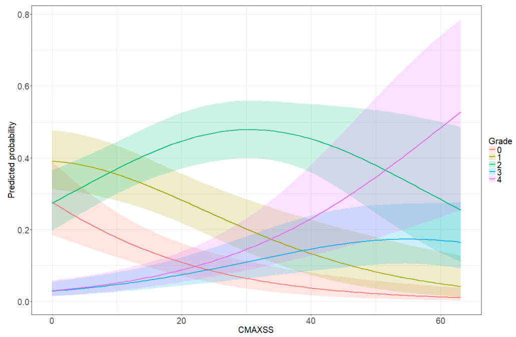
Distribution of training model parameters

Questions?

Prediction [1]

Why:

- Can use predictive plots to better understand the model.



- The final model can be used to predict new data. [Trial simulations]
- Can address ‘what if’ questions, such as influence of dose restrictions on risk etc.

Prediction [2]

How:

$$p(grade0) = 1 - \frac{e^{(exposure \times \beta_{exposure} + sex \times \beta_{sex} - \beta_{0|1})}}{\left(1 + e^{(exposure \times \beta_{exposure} + sex \times \beta_{sex} - \beta_{0|1})}\right)}$$

$$p(grade1) = 1 - \frac{e^{(exposure \times \beta_{exposure} + sex \times \beta_{sex} - \beta_{1|2})}}{\left(1 + e^{(exposure \times \beta_{exposure} + sex \times \beta_{sex} - \beta_{1|2})}\right)} - p(grade0)$$

And so on…

Or

R: predict function {stats,clm}

```
predict(cvfinal.pred, newdat1, se.fit=TRUE, interval=TRUE)
```

General methodology:

- Create new dataset with range of covariates.
- Predict probabilities of outcomes.

Prediction [3a]

<u>Predict</u>: generic function for predictions from the results of various model fitting functions.

- predict (object, DF, ...)
- Object – model object to predict from
- DF – new data frame to predict
- Other arguments include: *intervals* [provides CI], *level* [CI level, default = 95%], *se.fit* [provides standard errors], *type* [type of prediction 'prob' = probabilities (default), 'class' = response class (highest prob),...]

Model object:

```
cvfinal.pred<-clm(DV ~ EXP +SEX, data=full)
```

Predict DF:

```
# create data frame for what you want to predict into...
newdf <- data.frame(
  EXP = c(30,30),
  SEX = as.factor(c(1,2))
)
```

Remember to keep as factors

Prediction [3b]

Predict call for probs:

```
predict(cvfinal.pred, newdf,
        se.fit = TRUE,
        interval = TRUE,
        type = "prob") # predict probabilities
```

Output:

Probabilities for grade

SE estimates

Lower and Upper Intervals

```
$`fit`
           0          1         2         3         4
1 0.02742042 0.09588705 0.3861837 0.1651839 0.3253249
2 0.07373541 0.21051226 0.4614803 0.1084053 0.1458667

$se.fit
           0          1          2          3          4
1 0.01124271 0.03060969 0.05664006 0.04087654 0.07905026
2 0.02079135 0.03768186 0.04017236 0.02889366 0.03661755

$lwr
           0          1         2          3          4
1 0.01218934 0.05041136 0.2825801 0.09964859 0.19227415
2 0.04199402 0.14601259 0.3843297 0.06338488 0.08758881

$upr
           0        1         2         3         4
1 0.06051741 0.174833 0.5012335 0.2613104 0.4941221
2 0.12630555 0.293705 0.5405212 0.1792816 0.2330176
```

Predict call for class:

```
predict(cvfinal.pred, newdf,
        type = "class")   # predict classification
```

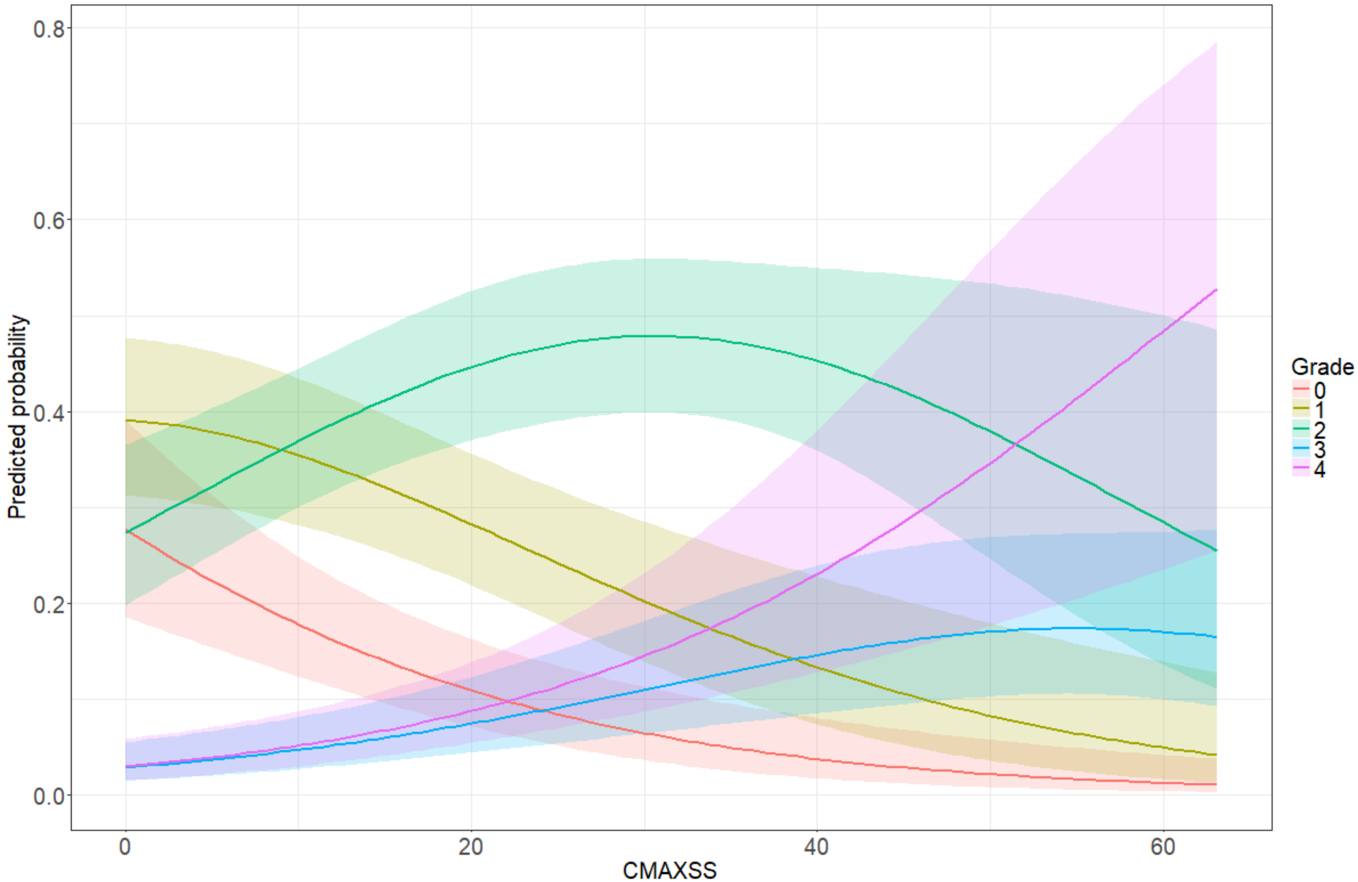
Output: predicted grade

```
$`fit`
[1] 2 2
Levels: 0 1 2 3 4
```

Note: if your DF which you are predicting contains your DV column (i.e. if you are predicting your dataset) the output is only the prediction of the grade in the DV column

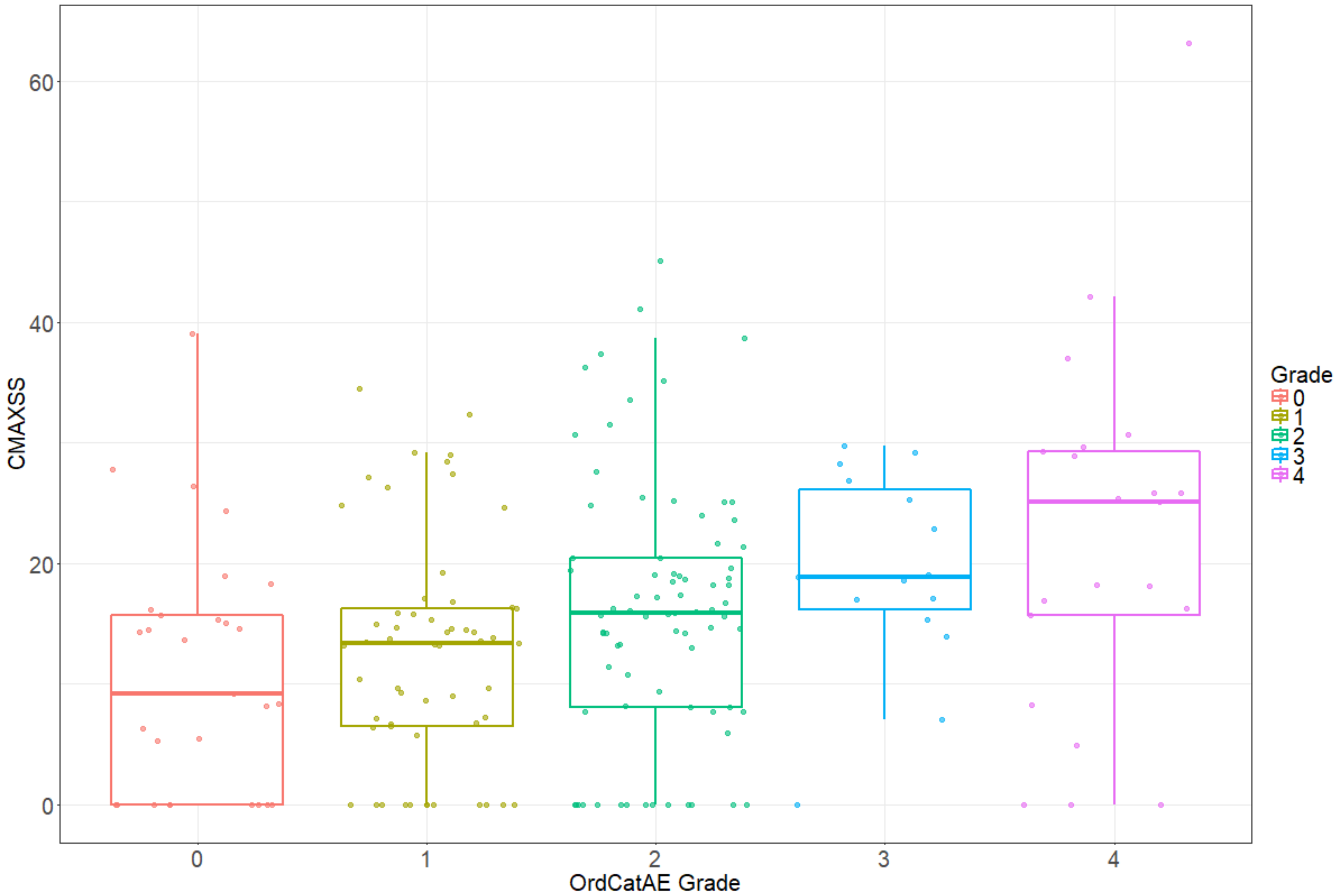
Prediction [4]

- Example – How does predicted probability for each grade change with exposure?



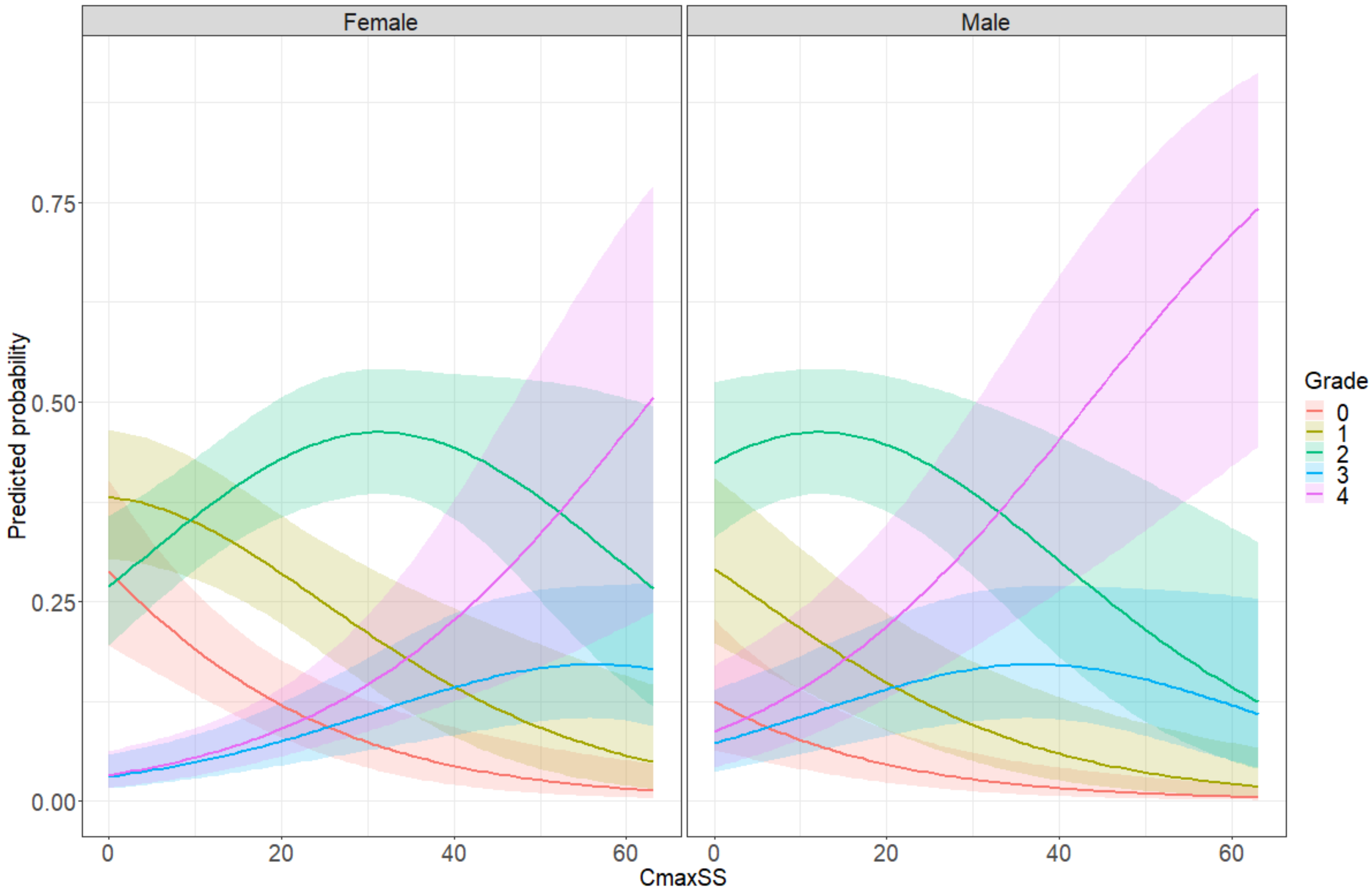
Mean & CI (90%)

Return to the raw data…



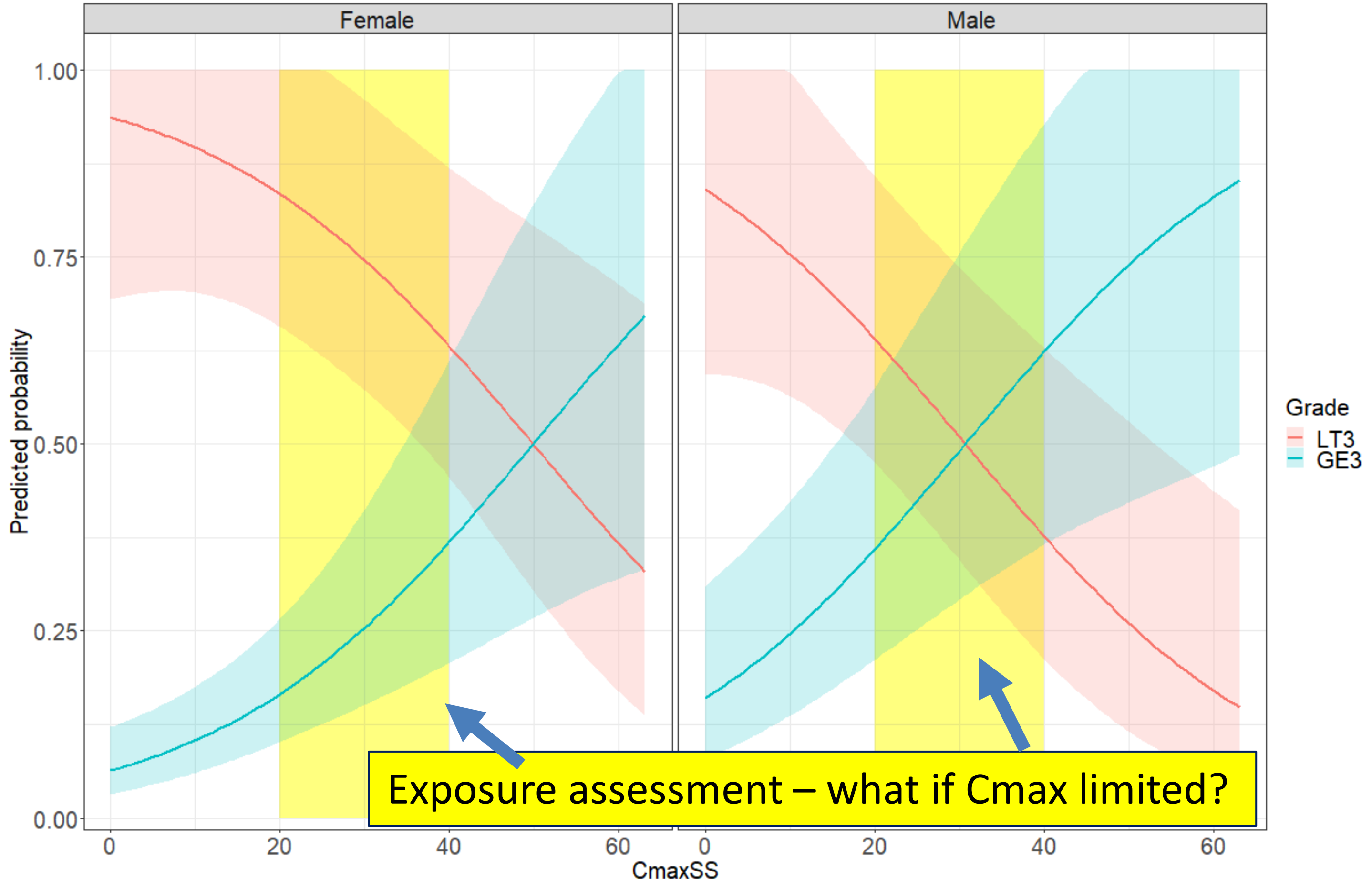
Prediction [5]

- Example – How does predicted probability for each grade change with exposure and gender?



Prediction [6]

- Example – How does predicted probability for grade < 3 change with exposure and gender?



Prediction [7]

Calculate change in risk for females of less than Grade 3 ?

- Prob = p(grade 0) + p(grade 1) + p(grade 2)
- Odds = Prob/(1-Prob)
- OddsRatio = $Odds_{female}/Odds_{male}$
  - OR: 2.82 for female to male having grade <3

- So, what does this look like?

Prediction [7b]

New DF

Probs

Odds

Odds Ratio

```
> ############################################################################
> ### create male, female and 30 or 60
> newdat3 <- data.frame(
+   EXP = c(0,0,5,5,30,30,45,45,60,60),
+   SEX = as.factor(rep(1:2,5)))
>
> #need to make sure that our datasets are factor for categoricals
>
> newdat <- cbind(newdat3, predict(cvfinal.pred, newdat3,
+                                  se.fit=TRUE, interval=TRUE))
> # names(newdat)
>
> newdat$pLT3 <- newdat$fit.0 + newdat$fit.1 + newdat$fit.2
> newdat$oddLT3 <- newdat$pLT3/(1-newdat$pLT3)
>
> # at CmaxSS = 60 odds ratio of female to male of grade LT 3
> OR.fem.mal.60 <- newdat$oddLT3[newdat$SEX==2&newdat$EXP==60]/
+   newdat$oddLT3[newdat$SEX==1&newdat$EXP==60]
>
> OR.fem.mal.60
[1] 2.823528
```

Note: can generate bootstrap interval for OR

Some thoughts on Verification and Validation

Validation – ‘the action of proving the validity or accuracy of something.’

For ordinal logistic regression the goal is to understand the risk but not necessarily the ‘classification’ of event.

Classification

- The ability of a model to correctly categorize
  - Machine learning applications (ROC, confusion tables)
  - Outcomes are distinct (is it a dog or cat?)
  - Predictors are strong (less applicable when the same input can have multiple outputs)
- This is different than prediction – which can differentiate the probability of multiple outputs.

Classification: 'Confusion' Matrix and Tables

T. Fawcett / Pattern Recognition Letters 27 (2006) 861–874

		True class	
		p	**n**
Hypothesized class	**Y**	True Positives	False Positives
	N	False Negatives	True Negatives
Column totals:		**P**	**N**

$$\text{fp rate} = \frac{FP}{N} \qquad \text{tp rate} = \frac{TP}{P}$$

$$\text{precision} = \frac{TP}{TP+FP} \qquad \text{recall} = \frac{TP}{P}$$

$$\text{accuracy} = \frac{TP+TN}{P+N}$$

$$\text{F-measure} = \frac{2}{1/\text{precision}+1/\text{recall}}$$

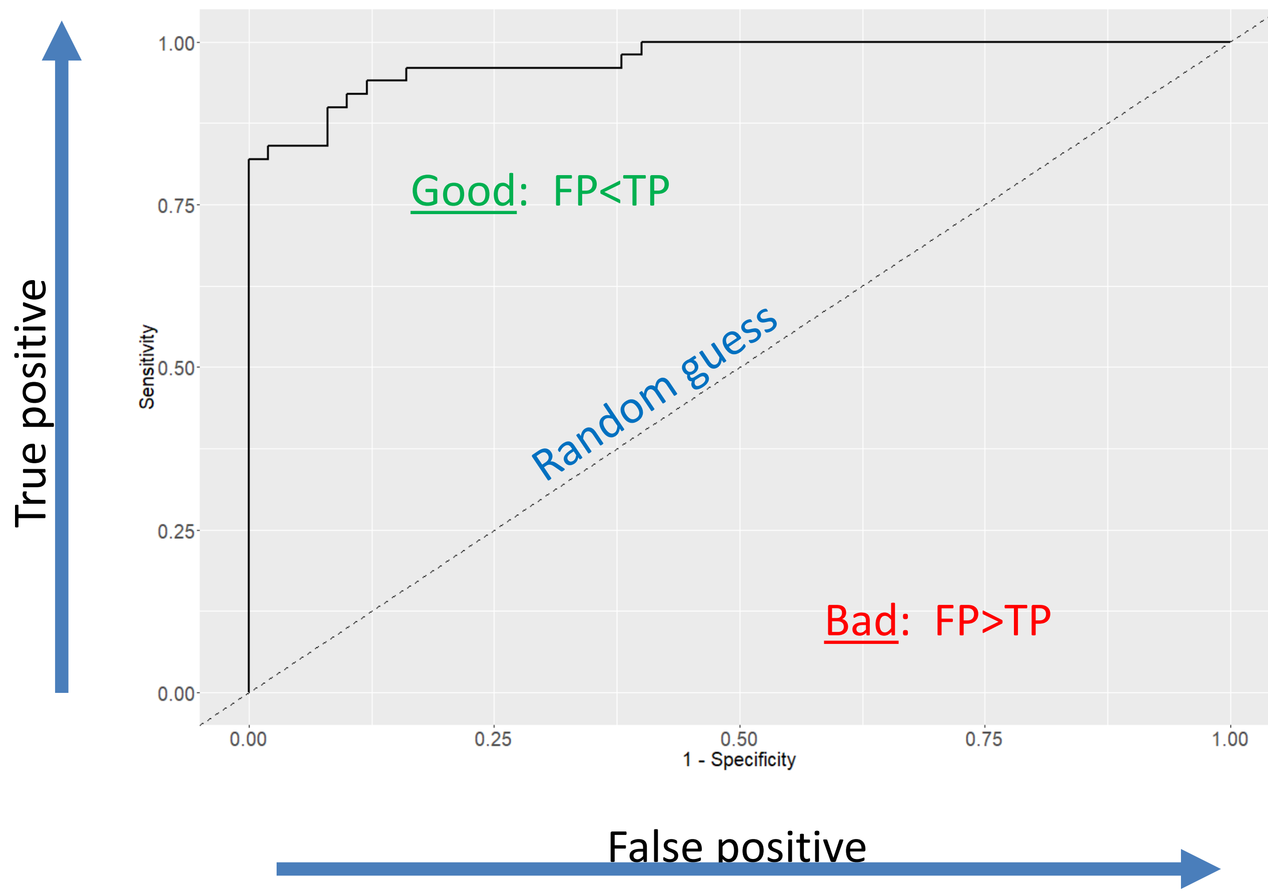
Fig. 1. Confusion matrix and common performance metrics calculated from it.

Sensitivity = tp rate

Specificity = 1- fp rate

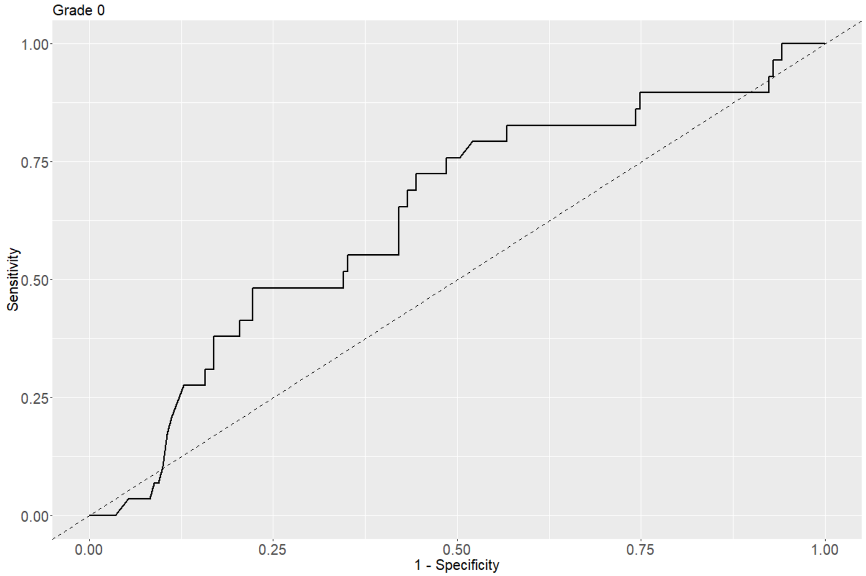
Classification: Receiver Operating Characteristic (ROC)

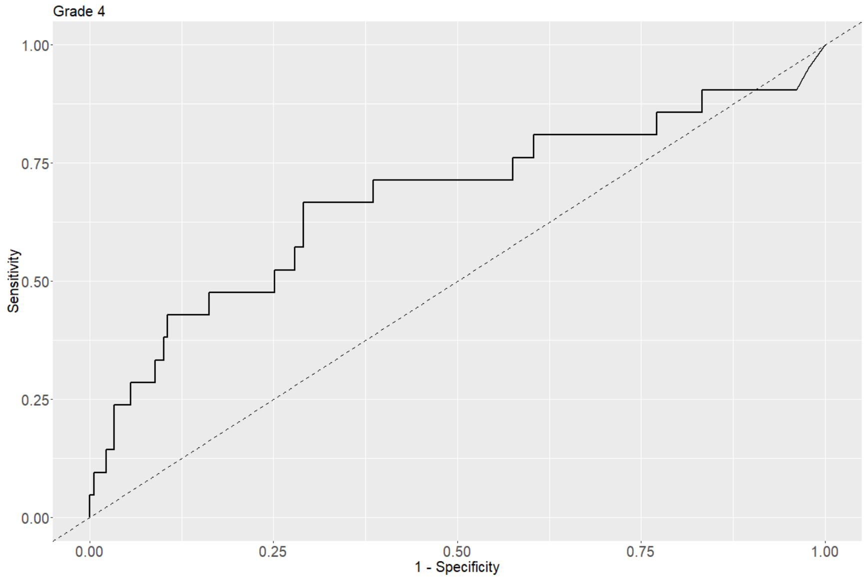
- ROC curve – shows diagnostic ability of binary classifier.



Classification: Cross-validation Output

- Review each grade - ROC

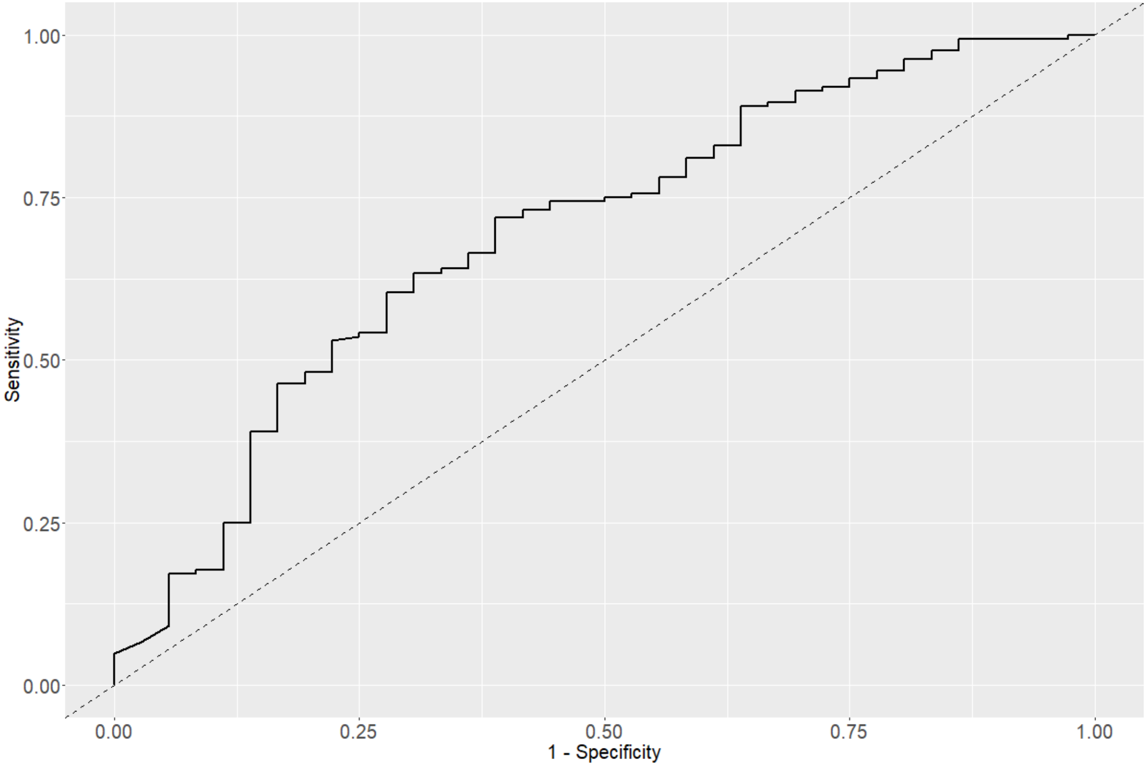




Thoughts???

Classification: Cross Validation Output

- What is ability to predict grade less than 3?
  – Compare known observations with grade < 3 with model predicted grade < 3.



	TruePos	TrueNeg	Total
PredPos	163	34	197
PredNeg	1	2	3
Total	164	36	200

	Rate
Sensitivity (TPR)	99.4
Specificity (TNR)	5.6
False Positive	94.4
False Negative	0.6

Classification: Cross-validation Output

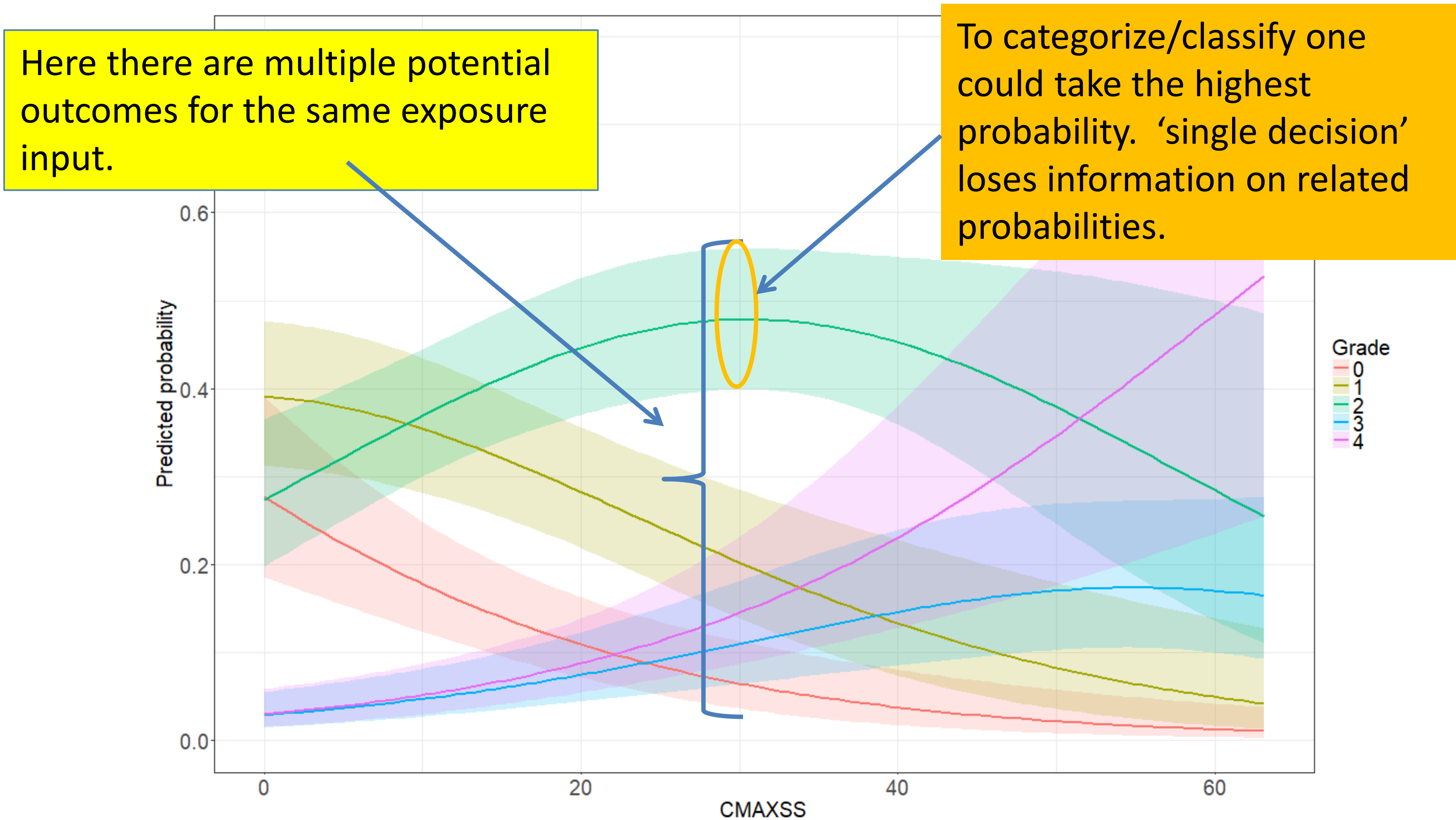
- Review each grade - matrix

	True Grade 0	True Grade 1	True Grade 2	True Grade 3	True Grade 4	Totals
Pred Grade 0	0	0	0	0	0	0
Pred Grade 1	11	17	13	1	3	45
Pred Grade 2	18	40	64	14	16	152
Pred Grade 3	0	0	0	0	0	0
Pred Grade 4	0	0	1	0	2	3
Totals	29	57	78	15	21	200
True Positive Rate	0.00%	29.8%	82.1%	0.00%	9.52%	100%

???

Classification: predicted probability

Think again on this predicted probability plot.



Logistic regression is not necessarily a good classification method.

Internal or External Validation

- Validation of a model requires prediction from the model into data which was *not used* during the model building.
- Proposed: either set aside data from the model process or use an external study. With same model structure ➔ same parameters?
- Internal methods: bootstrap etc. will re-use same dataset multiple times to re-estimate parameters.

Thoughts…

- Bootstrap (and other diagnostics like CDD) are easy to assess and provide understanding on parameters and model. Verification.
- Prediction of risk, not necessarily classification of grade.
- Predict function allows further exploration of model. [very open ended…]

Hands on

- Bootstrap and Predictive plots

References

- Davison & Hinkley, Bootstrap methods and their application, 1997 ISBN-10: 0521574714
- Efron & Tibshirani, An introduction to the bootstrap, 1993 ISBN-10: 0412042312
- Harris & Stocker, Handbook of mathematics and computational science, 1998 ISBN 0-387-94746-9
- Anguita et al., The ‘K’ in K-fold Cross Validation, 2012 ESANN proceedings
- Arlot, A survey of cross-validation procedures for model selection, 2010 Stat Survey v4, p40-79
- Burman, A comparative study of ordinary cross-validation, v-fold cross-validation and the repeated learning-testing methods, 1989 Biometrika, v76 p503-514
- Fawcett, An introduction to ROC analysis, 2006 Pattern Recognition Letters v27 p861-674
- Fushiki, Estimation of prediction error by using K-fold cross-validation
- Hand & Till, A simple generalization of the area under the ROC curve for multiple class classification problems, 2001 Machine Learning v45 p171-186

- R packages: boot
- R packages: ordinal, rms
- R packages: plotROC