Supplemental Materials Table of Contents

1 – Supplemental methods: expanded methodologic explanation of machine learning algorithms and metrics

2 – Supplemental table 1: all LASSO-selected metabolites based on CKD etiology (see attached excel file)

3 - Supplemental table 2: FSGS hyperparameter tuning analysis

4 – Supplemental table 3: Obstructive uropathy hyperparameter tuning analysis

5 – Supplemental table 4: Table of all metabolites included for pathway enrichment analysis (see attached excel file)

6 – Supplemental table 5: Feature importance rankings in different machine learning approaches

7 – Supplemental: Annotated machine learning sample code

Supplemental methods

Untargeted metabolomic profiling by Metabolon

For compound identification, Metabolon maintains a library based on authenticated standards that contain the retention time/index, mass to charge ratio, ad chromatographic data on all molecules present in the library. Furthermore, biochemical identifications are based on three criteria: retention index within a narrow window of proposed identification, accurate mass match to the library +/- 0.005 amu, and the MS/MS forward and reverse scores between the experimental data and the authenticated standards. More than 3300 commercially available purified standard compounds have been acquired and registered.

Lasso penalized logistic regression

Lasso is a penalized logistic regression model in which input features that do not significantly contribute to model performance have their coefficient estimates shrink towards zero by imposing a shrinkage parameter lambda. At optimized lambda with minimized misclassification error, there will be a number of input features with non-zero coefficient estimates. All features' coefficients will have downward penalties applied in this algorithm. Thus, Lasso is a more restrictive approach that will identify a smaller panel of metabolites with stronger signals. Those input features were selected for further analysis.

Random forest (RF)

RF is an aggregated tree-based model that randomly samples n-number of metabolites at branch points to determine classification. The number of metabolites (hyperparameter mtry) sampled at each branch point was determined based on the square root of total input metabolites (n Lasso-selected metabolites). In R *caret*, RF grows n-number of trees until model accuracy plateaus. The trees are aggregated into a final model. Metabolite importance was determined by how the exclusion of each metabolite decreased model accuracy.

Support vector machine (SVM)

Linear kernel SVM maps input data (Lasso-selected metabolites) to a high-dimensional space and determine the optimal hyperplane for binary classification (CKD etiology). The cost (C) hyperparameter is a penalization factor for misclassification, where a lower C is more generalizable. C was set at 1 to favor generalizability. Metabolite importance was determined by the coordinates of each data point's orthogonal vector to the hyperplane (weight).

Extreme gradient boosting (XGB)

XGB applies regression models in sequence (boosting) to minimize misclassification error of the preceding models, and then assembles them into an aggregated model. The number of rounds is the number of trees that XGB grows. Max depthcontrols how many times XGB will boost within each tree. Gamma is the regularization factor that penalizes multicollinear input features that do not improve model performance. These hyperparameters were manually set to favor generalizability and limit overfitting. Metabolite importance was determined by improvement in accuracy/gain as how each individual metabolite contributed to the model's overall performance.

Evaluation metrics

In ML, no-skill prediction is defined as a model that cannot discriminate between classes and would generate random or constant classes. ROC-AUC can overestimate performance in datasets with low case prevalence. The precision-recall (PR) curve evaluates ML performance and accounts of low case prevalence by plotting the positive predictive value (precision) against sensitivity (recall). No-skill ML models generating constant classes would plot as a horizontal line at the case prevalence rate on the PR curve. The F-1 score is a harmonic mean of the precision and recall values in which the magnitude is relative to model performance (no-skill = 0, perfect prediction = 1). The MCC is similar to the F-1 score, but includes directionality and accounts for true negatives (no-skill = 0, perfect negative prediction = -1, perfect positive prediction = 1).

F1-score = 2* (precision * recall)/(precision + recall)

= (true positives)/[true positives + 0.5*(false positives + false negatives)]

MCC = [(true positives * true negatives) – (false positives * false negatives)]/

Square-root[(true positives + false positive) * (true positives + false negatives) *(true negatives + false positives) * (true negatives + false negatives)]

Supplemental Table 2: ML model hyperparameter tuning analysis for FSGS

Support Vector Machine		
Hyperparameter	**Default**	**Tuned**
C (cost)	1	1 (range 1-20, by 1)
Metric	**Default**	**Tuned**
ROC-AUC	0.93	No changes in performance metrics
PR-AUC	0.60	
F1-score	0.43	
MCC	0.44	
Top 10% metabolites	**Default**	**Tuned**
	sphingomyelin (d18:1/18:1, d18:2/18:0)	No changes in metabolites detected
	1-(1-enyl-palmitoyl)-2-arachidonoyl-GPC (P-16:0/20:4)	
	sphingomyelin (d18:2/24:2)	
	1-(1-enyl-palmitoyl)-2-palmitoyl-GPC (P-16:0/16:0)	
	glycosyl ceramide (d18:2/24:1, d18:1/24:2)	
	glycosyl-N-behenoyl-sphingadienine (d18:2/22:0)	
Random forest		
Hyperparameter	**Default**	**Tuned**
Number of metabolites sampled at each branch point	8	14 (range 5-15, by 1)
Number of trees	500	500
Metric	**Default**	**Tuned**
ROC-AUC	0.89	0.89
PR-AUC	0.50	0.49
F1-score	0.50	0.53
MCC	0.48	0.50
Top 10% metabolites	**Default**	**Tuned**
	sphingomyelin (d18:1/18:1, d18:2/18:0)	No changes in metabolites detected
	1-arachidonoyl-GPI (20:4)	
	1-(1-enyl-palmitoyl)-2-palmitoyl-GPC (P-16:0/16:0)	
	1-(1-enyl-palmitoyl)-2-arachidonoyl-GPC (P-16:0/20:4)	
	6-bromotryptophan	
	homoarginine	
Extreme gradient boosting		
Hyperparameter	**Default**	**Tuned**
Number of rounds	100	100 (range 10-100, by 10)
Max depth	6	9 (range 5-10, by 1)
gamma	0	0 (range 0-1, by 0.5)
Metric	**Default**	**Tuned**
ROC-AUC	0.92	0.91
PR-AUC	0.55	0.55
F1-score	0.49	0.48
MCC	0.48	0.46
Top 10% metabolites	**Default**	**Tuned**
	1-(1-enyl-palmitoyl)-2-arachidonoyl-GPC (P-16:0/20:4)	sphingomyelin (d18:1/18:1, d18:2/18:0)
	palmitoyl-arachidonoyl-glycerol (16:0/20:4) [1]	homoarginine

	homoarginine	**1-(1-enyl-palmitoyl)-2-palmitoyl-GPC (P-16:0/16:0)**
	sphingomyelin (d18:1/18:1, d18:2/18:0)	**sphingomyelin (d18:2/24:2)**
	1-arachidonoyl-GPI (20:4)	palmitoyl-arachidonoyl-glycerol (16:0/20:4) [1]
	2-hydroxyarachidate	**urate**

We performed hyperparameter tuning analysis for FSGS to determine if ML model performance (i.e., ROC-AUC, PR-AUC, F1-score, and MCC) and metabolite signals detected (top10% metabolites) would be significantly different between default and hyperparameter-tuned ML models. Analysis was performed on the entire CKiD cohort (n=702) on the Lasso-selected metabolites for FSGS (n=56). Hyperparameter tuning was performed with a grid-search approach, with the ranges and increments of hyperparameters tested reported in parentheses. The best-tuned model was chosen based on maximized training prediction accuracy.

In SVM, the cost hyperparameter (C) is the penalization factor for misclassification. In RF, the number of metabolites sampled at each brand point (mtry) at default is determined by the square root of the number of input features, while the number of trees to grow has been optimized in the R ‘caret’ package source code to be capped once model performance plateaus. There was no significant differences in performance metrics or metabolite signals detected between default and hyperparameter-tuned models for both SVM and RF.

In XGB, the number of rounds is comparable to the number of trees in RF. The max depth controls XGB model depth. Gamma is the regularization factor that penalizes large coefficients that do not improve model performance. Default and hyperparameter-tuned model did not differ significantly in the performance metrics. There were some differences in important feature rankings (highlighted and bolded). Lipid subpathway metabolites were consistently implicated between the default and hyperparameter-tuned models (sphingomyelin, plasmalogen, lysophospholipid, and ceramide subpathways). 4 out of the 6 different metabolites were ultimately implicated as associated with FSGS in our final analyses.

Supplemental Table 3: ML model hyperparameter tuning analysis for OU

Support Vector Machine		
Hyperparameter	**Default**	**Tuned**
C (cost)	1	2 (range 1-20, by 1)
Metric	**Default**	**Tuned**
ROC-AUC	0.86	0.86
PR-AUC	0.54	0.54
F1-score	0.57	0.56
MCC	0.47	0.47
Top 10% metabolites	**Default**	**Tuned**
	imidazole propionate	No changes in metabolites detected
	trans-urocanate	
	4-methoxyphenol sulfate	
	5,6-dihydrothymine	
Extreme gradient boosting		
Hyperparameter	**Default**	**Tuned**
Number of rounds	100	80 (range 10-100, by 10)
Max depth	6	9 (range 5-10, by 1)
gamma	0	1 (range 0-1, by 0.5)
Metric	**Default**	**Tuned**
ROC-AUC	0.80	0.80
PR-AUC	0.47	0.47
F1-score	0.48	0.47
MCC	0.37	0.36
Top 10% metabolites	**Default**	**Tuned**
	imidazole propionate	No changes in metabolites detected
	trans-urocanate	
	4-methoxyphenol sulfate	
	glycerol 3-phosphate	
Random forest		
Hyperparameter	**Default**	**Tuned**
Number of metabolites sampled at each branch point	7	15 (range 5-15, by 1)
Number of trees	500	500
Metric	**Default**	**Tuned**
ROC-AUC	0.74	0.73
PR-AUC	0.40	0.39
F1-score	0.43	0.42
MCC	0.29	0.28
Top 10% metabolites	**Default**	**Tuned**
	trans-urocanate	No changes in metabolites detected
	imidazole propionate	
	N-acetylkynurenine	
	glycerol 3-phosphate	

We performed hyperparameter tuning analysis for OU to determine if ML model performance and metabolite signals would be significantly different between default and hyperparameter-tuned ML models. Analysis was performed on the entire CKiD cohort (n=702) on the 43 Lasso-selected metabolites. There were slight differences in optimized hyperparameters for all ML models, but there were no significance changes in performance metrics or metabolite importance weighting.

Supplemental Table 5: Feature importance rankings comparison

FSGS – Logistic regression – metabolites meeting FDR threshold		
Pathway	**Metabolite**	**P-value**
Sphingomyelin	**Sphingomyelin (d18:1/18:1, d18:2/18:0)**	**2.95e-10**
Plasmalogen	**1-(1-enyl-palmitoyl)-2-palmitoyl-GPC (P-16:0/16:0)**	**8.19e-9**
Plasmalogen	**1-(1-enyl-palmitoyl)-2-arachidonoyl-GPC (P-16:0/20:4)**	**2.32e-8**
Hexosylceramide	Glycosyl ceramide (d18:2/24:1, d18:1/24:2)	3.09e-8
Tryptophan	6-bromotryptophan	7.00e-8
Sphingomyelin	**Sphingomyelin (d18:2/24:2)**	**9.52e-8**
Hexosylceramide	Glycosyl-N-behenoyl-sphingadienine (d18:2/24:1(2OH))	1.42e-7
Tryptophan	Indole-3-carboxylate	2.14e-6
Tryptophan	Indoleproprionate	4.51e-6
Fatty acid	Hydroxyl-CMPF*	1.54e-5
Lysophospholipid	**1-arachidonoyl-GPI* (20:4)**	**1.78e-5**
Glutamate	**N-acetyl-aspartyl-glutamate (NAAG)**	**2.76e-5**
Urea cycle	Homoarginine	3.26e-5
Tyrosine	Thyroxine	3.45e-5
Acetylacetate	Phenylacetylglycine	4.10e-5
Diacylglycerol	**Palmitoyl-arachidonoyl-glyecerol (16:0/20:4) [1]**	**4.31e-5**
Glutamate	Beta-citrylglutamate	6.77e-5
Lysine	N,N-dimethyl-5-aminovalerate	7.94e-5
Bile acid	isoursodeoxycholate	1.54e-4
Tyrosine	N-formylphenylalanine	2.83e-4
Lysophospholipid	1-palmitoyl-GPG (16:0)*	3.35e-4
Pantothenate/CoA	Pantothenate (Vitamin B5)	5.39e-4
Fatty acid	2-hydroxyarachidate*	5.44e-4
Sterol	7-HOCA	5.82e-4
Urea cycle	N-acetylcitrulline	6.50e-4
FSGS – Support vector machine – top 10th percentile metabolite ranked weightings in 1 training iteration example, ROC-AUC = 0.93, PR-AUC = 0.50		
Pathway	**Metabolite**	**Ranked weight**
Sphingomyelin	**Sphingomyelin (d18:1/18:1, d18:2/18:0)**	**100.00**
Plasmalogen	**1-(1-enyl-palmitoyl)-2-palmitoyl-GPC (P-16:0/16:0)**	**97.29**
Tryptophan	**Indole-3-carboxylate**	92.74
Sphingomyelin	**Sphingomyelin (d18:2/24:2)**	**92.63**
Plasmalogen	**1-(1-enyl-palmitoyl)-2-arachidonoyl-GPC (P-16:0/20:4)**	**88.36**
Glutamate	**N-acetyl-aspartyl-glutamate (NAAG)**	**83.43**
FSGS – Random forest – top 10th percentile metabolite ranked weightings in 1 training iteration example, ROC-AUC = 0.88, PR-AUC = 0.49		
Pathway	**Metabolite**	**Ranked weight**
Sphingomyelin	**Sphingomyelin (d18:1/18:1, d18:2/18:0)**	**100.00**
Plasmalogen	**1-(1-enyl-palmitoyl)-2-palmitoyl-GPC (P-16:0/16:0)**	**50.96**
Glutamate	beta-citrylglutamate	49.25
Lysophospholipid	1-arachidonoyl-GPI (20:4)	46.30
Plasmalogen	**1-(1-enyl-palmitoyl)-2-arachidonoyl-GPC (P-16:0/20:4)**	**45.71**
Pantothenate/CoA	**Pantothenate (Vitamin B5)**	**42.54**
FSGS – Extreme gradient boosting – top 10th percentile metabolite ranked weightings in 1 training iteration example, ROC-AUC = 0.91, PR-AUC = 0.40		
Pathway	**Metabolite**	**Gain**
Fatty acid	2-hydroxyarachidate	0.050
Tryptophan	Indole-3-carboxylate	0.048
Glutamate	**N-acetyl-aspartyl-glutamate (NAAG)**	**0.044**
Acetylacetate	Phenylacetylglycine	0.043
Plasmalogen	**1-(1-enyl-palmitoyl)-2-palmitoyl-GPC (P-16:0/16:0)**	**0.038**
Plasmalogen	**1-(1-enyl-palmitoyl)-2-arachidonoyl-GPC (P-16:0/20:4)**	**0.036**

This table shows the metabolites that met the significance/importance designations per modeling approach for FSGS: Bonferroni threshold for LR, and top 10th percentile ranked weight/gain in 1 out of 10 training iterations of SVM and XGB respectively. Metabolites that were ultimately implicated as defined by our analytic schematic are shown to be similarly highly ranked across all 3 modeling approaches.

Supplemental: annotated machine learning sample code – Annotations are denoted by italicized text following “###”. All analyses were performed in R studio version 4.0.5. All R packages utilized are bolded and available from the Comprehensive R Archive Network (CRAN) repository. Package versions are annotated. This sample code has been uploaded to a GitHub repository (https://github.com/leeam-chop/CKiD_rcode/blob/a450ca7e64011750e928b0c90ca98e66d248f55e/etiologies_metabolomics).

Consistent with CKD Biocon and CKiD data sharing policies, a de-identified dataset can be provided for replication purposes through a data use agreement (DUA) between the investigator’s institution and Johns Hopkins University. Investigators interested in accessing the de-identified dataset should contact Judith Jerry-Fluker at the Kidney Disease in Children Data Management and Analysis Center (KIDMAC): jjerry@jhu.edu.

```
###================================1. Load Required Packages=====================================
library(readr) ###version==1.4.0
library(dplyr) ###version==1.0.6
library(caret) ###version==6.0-88
library(e1071) ###version==1.7-7
library(MLeval) ###versio== 0.3
library(xgboost) ###version==1.4.1.1
library(stringr) ###version==1.4.0
library(car) ###version==3.0-10

###================================2. Load Data================================================
###Data has undergone QC procedures as detailed in the manuscript
###Categorical variables have been assigned as factors: hypertension, sex, ACEi/ARB usage, race
###Each primary etiology is coded as a binary factor, ex: fsgs (FSGS =1, not FSGS =0)
###Proteinuria and eGFR have been log-2-transformed

###================================3. Feature selection using Lasso regression=======================
lasso <- select(data, c(id, etiology, age, sex, race, eGFR, proteinuria, hypertension, ACEi/ARB usage)
levels(lasso#etiology) <- c(“negative, “positive”)
trctrl <- trainControl(method=”repeatedcv”, number=10, repeats=3, classProbs=TRUE, savePredictions=TRUE)
trgrid <- expand.grid (alpha=1, lambda=seq(0.01, 0.1, by=0.01)
set.seed(1)
model <- train(etiology~.,
        data=lasso,
        method=”glmnet”,
        metric=”Accuracy”,
        trControl=trctrl,
        tuneGrid=trgrid,
        preProc=c(“zv”, “center”, “scale”))
model#bestTune
evalm(model, plots=”r”)
coef <- coef(model$finalModel, model$bestTune$lambda)[,1]
coef <- data.frame(coef)
coef <- abs(coef)
sum(coef != 0)
write.csv(coef, file=”coef.csv)
        ###identified features with non-zero coefficients at cross-validated, tuned lambda

###================================4. Subset data with Lasso-selected metabolites for each etiology======
###Use 10 unique seeds to repeat training process 10 times
set.seed(1)
```

```
levels(data_subset$etiology) <- c(“negative”, “positive”)
###80-20 train-test split
train <- data_subset %>% dplyr::sample_frac(0.8)
test <- dplyr::anti_join(data_subset, train, by=”id”)
###Remove subject ID from data for modeling
train <- train[,-1]
test <- test[,-1]
trctrl <- trainControl(method=”repeatedcv”, number=10, repeats=3, classProbs=TRUE, savePredictions=TRUE)

###================================5. Fit Support Vector Machine================================
svm <- train(etiology~.,
             data=train,
             method=”svmLinear”,
             trControl=trctrl)
pred <- predict(svm, newdata=test, type=”prob”)
evalm(data.frame(pred, test$etiology)
             ###assesses model performance on hold-out test split
varImp(svm)
             ###identifies 10% important metabolites
###repeat this with the 10 unique seedings

###================================6. Fit Extreme gradient boosting==============================
grid <- expand.grid(nrounds=100, max_depth=6, colsample_bytree=0.5, eta=0.1, gamma=0, min_child_weight =1)
xgb <- train(etiology~.,
             data=train,
             method=”xgbTree”,
             trControl=trctrl,
             tuneGrid=grid)
pred <- predict(xgb, newdata=test, type=”prob”)
evalm(data.frame(pred, test$etiology)
varImp(xgb)

###================================7. Fit Random forest==========================================
grid <- expand.grid(.mtry = ###square root of the number of lasso-selected metabolites per etiology)
rf <- train(etiology~.,
             data=train,
             method=”rf”,
             tuneGrid=grid,
             trControl=trctrl)
pred <- predict(rf, newdata=test, type=”prob”)
evalm(data.frame(pred, test$etiology)
varImp(rf)
```