

**Tutorial Title: Metagenomics analysis of microbiome data using machine learning approaches using MATLAB.**

**Lindsay Hopson<sup>1</sup>, John David<sup>2</sup>, Atin Basuchoudhary<sup>2</sup>, Stephanie Singleton<sup>1</sup>, Raja Mazumder<sup>1</sup>**

George Washington University<sup>1</sup>  
Virginia Military Institute<sup>2</sup>

We have chosen to apply the Creative Commons Attribution 4.0 International (CC BY 4.0) license to this tutorial. This means that you are free to copy, distribute, display and make commercial use of these databases in all legislations, provided you give us credit.

## **PURPOSE**

The purpose of this tutorial is to demonstrate machine-learning analysis for metagenomics data in MATLAB. This tutorial is for users with little to no MATLAB experience but have a basic understanding of machine learning concepts, such as data preparation, machine learning algorithms, and visualizations. We advise beginners to familiarize themselves with the previously mentioned topics prior to attempting this tutorial. Refer to the Appendix, beginning on page 7, for helpful MATLAB and machine learning resources.

## SUMMARY

The metagenomics data used in this tutorial was generated from bioinformatics analysis of fecal samples collected from wild-type (WT) and transforming growth factor-beta-signaling-deficient (TGF- $\beta$ ) mice at three different time points; before treatment (BT), during treatment (DT), and after treatment (AT) with Fluorouracil (5-FU; chemotherapeutic drug) or phosphate buffered saline (PBS) control. The organisms identified in these samples and their relative abundances are available in the Excel file "MGPC\_BMM\_CRC\_Mouse\_Microbiome\_Final3.xlsx". Using this data, the objective is to use MATLAB and machine learning approaches to answer the following questions:

- 1) *Is there any signal differentiating between TGF- $\beta$  and WT before treatment?*
- 2) *If there is signal, what are the important predictors?*
- 3) *Is there any signal differentiating between WT before treatment and WT after treatment with 5-FU (WT\_F\_BT vs WT\_F\_AT)?*

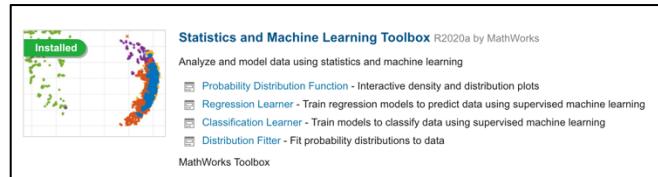
## DOWNLOAD REQUIRED MATERIALS

- MATLAB (version R2020a)  
(<https://seascf.seas.gwu.edu/install-matlab>)
- MGPC\_BMM\_CRC\_Mouse\_Microbiome\_Final3.xlsx  
(<https://drive.google.com/file/d/1MLy1u3CDbEtVEKkd9eTMKD00nutxdLAY/view?usp=sharing>)
- ensemble\_bagged.m  
(<https://drive.google.com/file/d/1HCdTX-t3qaPRYfMV3xUzrlJ9hUhzJ1hr/view?usp=sharing>)

## STEP-BY-STEP INSTRUCTIONS

### 1. Installing the Statistics and Machine Learning Toolbox

- a. For first time users, once you download MATLAB and open the program, it will provide the user with some toolbox options to download. Here you can click on the Statistics and Machine Learning Toolbox.
- b. If MATLAB is already installed and opened on your computer, the user can download different toolboxes by selecting APPS >> Get More Apps. A new MATLAB window will pop up. The user can then type in the search bar "Statistics and Machine Learning Toolbox", select the Statistics and Machine Learning Toolbox, and select the blue Install button. If the toolbox is already installed, there will be a green tab that says Installed (as shown on the right).



## 2. Data Scrubbing

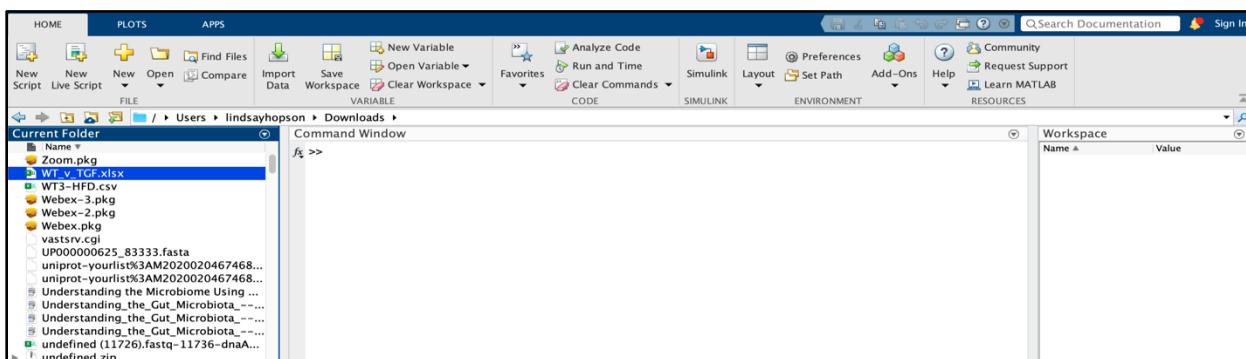
Before the data is uploaded into MATLAB, the data will have to be modified to remove irrelevant information, reformatted, and/or transformed based on the specific aims of the analysis. The data in `MGPC_BMM_CRC_Mouse_Microbiome_Final3.xlsx` is modified to help answer the first question. (**Is there any signal differentiating between TGF and WT before treatment?**)

- Open `MGPC_BMM_CRC_Mouse_Microbiome_Final3.xlsx` and Save As “`WT_v_TGF.xlsx`” in the Downloads folder. Delete the `README` sheet. Remove all the following columns from the table: `Lineage`, `GenBank_Reference`, `WT_F_DT`, `WT_P_DT`, `WT_F_AT`, `WT_P_AT`, `TGF_F_DT`, `TGF_P_DT`, `TGF_F_AT`, and `TGF_P_AT`.
- Transpose the rows and columns by first selecting all the data (including the row and column names). Next, right click a cell in the first column right below the data. Select `Special paste >> Transpose`. The transposed data will paste underneath the original formatted data. Next, delete all the original formatted data located above the transposed data. The new final table should now have all rows represented as samples and columns represented as bacteria species (MATLAB-friendly formatting shown below).
- Modify the `Genus_Species_Strain` column. Change the column name to “`Mouse_Type`”. For all subsequent data under `Mouse_Type`, reduce the specificity of the mouse type to “`WT`” and “`TGF`” (as shown below). Save this file.

1	<b>Mouse_Type</b>	<i>Helicobacter_typhlonius</i>	<i>Bifidobacterium_pseudo</i>	<i>Bacteroides_caecumuris</i>	<i>Bifidobacterium_pseudo</i>	<i>Faecalibaculum</i>
2	WT	0.17178828	0.17086862	0.07341315	0.07004965	0.06
3	WT	0.03324217	0.14408587	0.05635981	0.06499063	0.06
4	WT	0.10039678	0.18286385	0.05463321	0.07752780	0.06
5	WT	0.07963623	0.00000000	0.20628582	0.00001558	0.00
6	WT	0.07779820	0.000000306	0.23050484	0.00000000	0.00
7	WT	0.15377776	0.00000000	0.24184838	0.00000329	0.00
8	WT	0.15377776	0.00000000	0.24184838	0.00000329	0.00

## 3. Uploading Data into MATLAB

- Access the Excel file, `WT_v_TGF.xlsx`, in MATLAB by selecting the  symbol (Browse for Folder). Select the `Downloads` folder >> Open. All the files in the `Downloads` folder are listed in the panel on the far left (shown below).

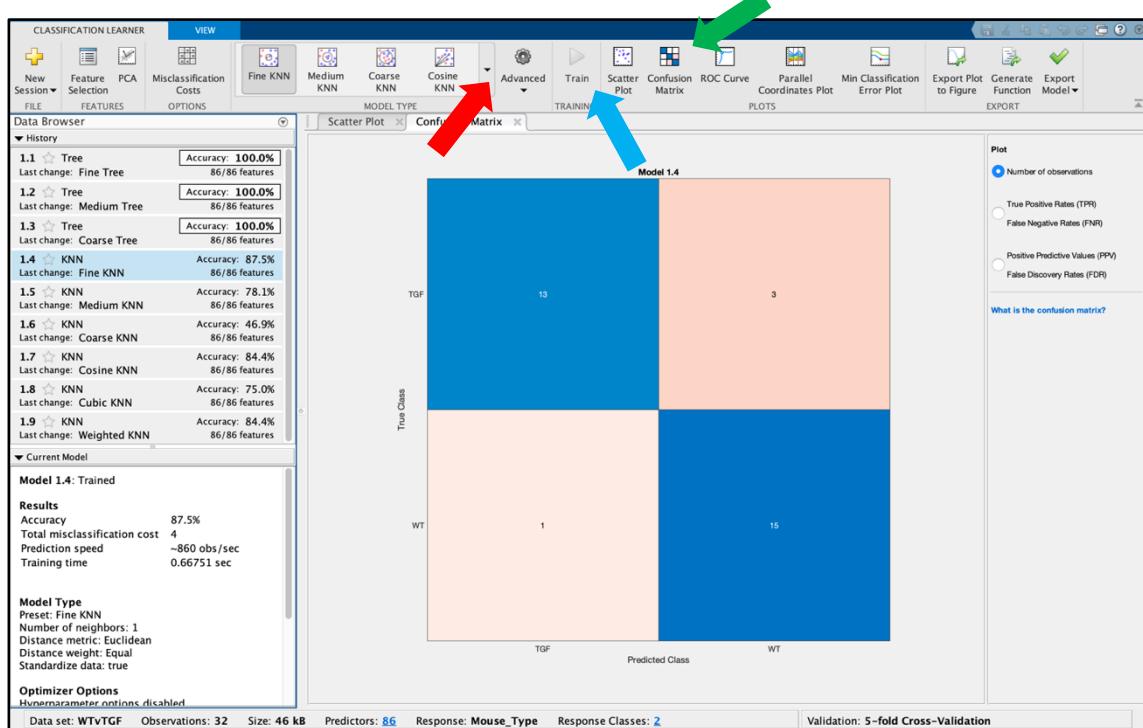


- Select `WT_v_TGF.xlsx` under Current Folder and drag the file into the Command Window. An import wizard (shown below) will appear. Under `Output Type:`, make sure that `Table` is selected. Make sure that all the data (only data values; no column names, or empty cells selected) are selected/highlighted. Next, select `Import Selection`. The `WTvTGF` table will then appear in the Workspace on the far right.

	Mouse_Type	WT	0.1718	0.1709	0.0734	0.0700	0.0628	0.0555	0.0486	0.0357	0.0349	0.0249	0.0235	0.0224	0.0213	0.0145	0.0138	0.0129	0.0124	0.0
1	WT	0.0332	0.1441	0.0564	0.0650	0.0650	0.0633	0.0430	0.0821	0.0585	0.0200	0.0432	0.0370	0.0229	0.0291	0.0225	0.0123	0.0123	0.0	
2	WT	0.1004	0.1829	0.0546	0.0775	0.0646	0.0416	0.0532	0.0573	0.0516	0.0130	0.0334	0.0328	0.0084	0.0186	0.0216	0.0180	0.0087	0.01	
3	WT	0.0796	0	0.2063	1.5577e...	0.0049	0.1462	1.2745e...	0.0024	7.0805e...	0.0281	9.1622e...	1.6993e...	0.0052	2.7331e...	0.0010	1.1895e...	0.0980		
4	WT	0.0778	3.0579e...	0.2305	0	0.0076	0.1650	2.0386e...	5.5960e...	6.7275e...	0.0561	2.6910e...	7.8487e...	0.0201	1.0499e...	2.3037e...	5.6062e...	0.0814	8.154	
5	WT	0.1538	0	0.2418	3.2858e...	8.8715e...	0.0762	1.9715e...	0.0049	7.2944e...	0.0361	0.0017	2.1686e...	0.0135	7.2484e...	0.0021	9.2001e...	0.0999	7.228	
6	WT																			
7	WT																			

#### 4. Determining the Best Classification Model to Detect and Predict Signal Differences in Mouse Type

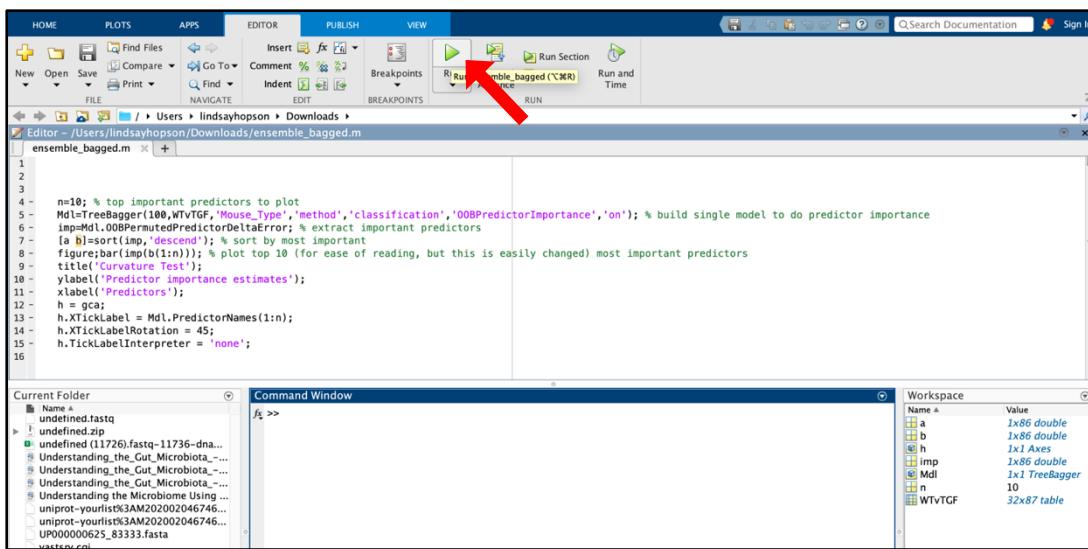
- a. Select the APPS tab at the top of the page. Select Classification Learner . In the Classification Learner GUI, select New Session >> From Workspace. In the New Session window under Data Set Variable, make sure the WTvTGF table is selected. Under Response, make sure that the From data set variable button is selected and Mouse\_Type is selected in the drop-down menu (this is our response variable we want to predict). Under Predictors, Mouse\_Type is unselected, and all the bacteria strains are selected. The select Start Session.
- b. In the Classification Learner tab, click the dropdown arrow (red arrow shown below). Under GET STATEDRED and select All. Then select Train (blue arrow shown below). MATLAB will then load. During this time, MATLAB is testing each algorithm on the data to generate the best predictive model. In the panel on the left shows the different models generated and their percent accuracy. One can view the ROC or AUC graphs for the different models by selecting the ROC Curve or Confusion Matrix buttons (green arrow shown below). All three single decision trees performed with a 100% accuracy. This means there is enough signal to differentiate between WT mice and TFG-β mice.



## 5. Finding the Most Important Predictors

In this step we will use the MATLAB code from the downloaded materials (`ensemble_bagged.m`) to answer the second question. (If there is signal, what are the important predictors?) Click the link for further explanation on predictor importance <https://christophm.github.io/interpretable-ml-book/feature-importance.html>.

- To perform predictor importance, check the accuracy of the bagged tree-based models. Importance variables can only be analyzed using bagged-tree models in MATLAB. For our data, the Ensemble Bagged Trees model had an accuracy of 81.2%. Since this model has decent accuracy, we can have a stronger confidence that the computed important variables are actually important when classifying mouse type. If the Ensemble Bagged Trees model had a poor accuracy, we could still compute the important predictors, however, we would not have strong confidence that the computed important variables are actually important when classifying mouse type. This is because our sample size is very small and it would be unwise to draw any formal conclusions from the predictor importance computation.
- Leaving the Classification Learner GUI and returning to the main MATLAB page, double click the `ensemble_bagged.m` seen in the Current Folder panel on the left. Code will load into the Command Window (shown below). Next, select the Editor tab at the top of the MATLAB window. Select Run. You can also run the code by typing “`ensemble_bagged`” into the Command Window and then selecting Enter.



- After the code runs, the important predictors plot is displayed.

## 6. Data Scrubbing

Before the data is uploaded into MATLAB, the data will have to be modified to remove irrelevant information, reformatted, or transformed based on the specific aims of the analysis. The data in `MGPC_BMM_CRC_Mouse_Microbiome_Final3.xlsx` is modified to help answer the second question (Is there any signal differentiating between WT before treatment and WT after treatment of 5-FU?)

- Open `MGPC_BMM_CRC_Mouse_Microbiome_Final3.xlsx` and Save As “`WTbeforeAfterData.xlsx`” in the Downloads folder. Delete the README sheet. Delete all the following columns: `Lineage`, `GenBank_Reference`, `TGF_F_BT`, `TGF_F_DT`, `TGF_F_AT`, `TGF_P_BT`, `TGF_P_DT`, `TGF_P_AT`, `WT_P_DT`, and `WT_F_DT`.

- b. Transpose the rows and columns by first selecting all the data (including the row and column names) can copying in. Next, right click on a cell in the first column right below the data. Select Special paste >> Transpose. The transposed data will paste underneath the original formatted data. Next, delete all the original formatted data located above the transposed data. The new final table should now have all samples represented as rows and bacteria species represented as columns (MATLAB-friendly formatting shown below).
- c. Modify the Genus\_Species\_Strain column. Change the column name to “Treatment”. For all subsequent data under Treatment, specify if the mouse received treatment with Before Treatment (BT) or After Treatment (AT) (as shown below). Save the file.

	Treatment	Helicobacter_typhi	Bifidobacterium_ps	Bacteroides_caecim	Bifidobacterium_ps	Faecalibaculum_rod	Muribaculun
1	BT	0.14108306	0.00000497	0.13653949	0.00000332	0.00156206	0.250
2	BT	0.17048885	0.00000115	0.10586479	0.00000689	0.00010797	0.159
3	BT	0.17767090	0.00000000	0.08390862	0.00000270	0.00088437	0.049
4	BT	0.07080634	0.00000000	0.05861068	0.00004467	0.00880054	0.121
5	BT	0.06491338	0.00000185	0.04317964	0.00000555	0.04944164	0.136
6	BT	0.07475246	0.00000099	0.26610377	0.00000000	0.00077953	0.090
7	BT	0.02407023	0.00000000	0.12974076	0.00000359	0.13224191	0.168
8	BT						

## 7. Uploading Data into MATLAB

- a. Access the Excel file, WTbeforeAfterData.xlsx, in MATLAB by selecting the Home tab. In the Current Folder panel on the left, you should be able to locate the Excel file. Double click the file.
- b. An import wizard will appear. Under Output Type:, make sure that Table is selected. Make sure that all the data (only data values; no column names, or empty cells selected) is selected/highlighted. Next, select Import Selection. The WTbeforeAfterData table will then appear in the Workspace on the far right.

## 8. Determining the Best Classification Model to Detect and Predict Signal Differences in Treatment

- a. Select the APPS tab at the top of the page. Select Classification Learner  . In the Classification Learner GUI, select New Session >> From Workspace. In the New Session window, under Data Set Variable, make sure the WTbeforeAfterData table is selected. Under Response, make sure that the From data set variable button is selected, and Treatment is selected in the drop-down menu (this is our response variable we want to predict). Under Predictors, Treatment is unselected, and all the bacteria strains are selected. The select Start Session.
- b. Click the dropdown arrow and select All. Then select Train. MATLAB will then load. During this time, MATLAB is testing each algorithm on the data to generate the best predictive model. On the left-most panel shows the different models generated and their percent accuracy. One can view the ROC or AUC graphs for the different models by selecting the ROC Curve or Confusion Matrix buttons. Ensemble Subspace Discriminant model had a 100% accuracy. SVM and KNN models had an accuracy of ~85%. The Bagged Tree model had an accuracy of 85.7%. Though the Bagged Tree model demonstrated a descent accuracy, we should remain extremely critical of this number, as our sample size is very small, and it would be unwise to draw any formal conclusions from the predictor importance computation.

## 9. Statistical Significance Testing in R

To support the results of the computed important predictors, statistical significance was assessed on all top 5 important predictors.

- a. Determine the sample cohorts follow Gaussian distribution (normal distribution). Understanding the distribution is required in order to determine the type of significance that will be performed (parametric or nonparametric). Normality can be assessed using many different tools (i.e R, MATLAB, ect). In this tutorial, normality was assessed using R code found in the link (<http://www.sthda.com/english/wiki/normality-test-in-r#install-required-r-packages>). From the results of the normality test and visualizations of the distribution through Q-Q plot and density plots and cohort sample size, the distribution could not be concluded to be normally distributed.
- b. Mann-Whitney U test (nonparametric test) was performed on the top 5 important predictors in each pairwise comparison (<https://www.statmethods.net/stats/nonparametric.html>).

## APPENDIX

### Machine Learning Resources:

## YouTube Videos

- <https://www.youtube.com/watch?v=G7fPB4OHkys>
- <https://www.youtube.com/watch?v=h0e2HAPTGF4>

## Books

- Machine Learning for Absolute Beginners (Second Edition) by Oliver Theobald [https://www.amazon.com/gp/product/1549617214/ref=ppx\\_yo\\_dt\\_b\\_asin\\_title\\_o00\\_s00?ie=UTF8&psc=1](https://www.amazon.com/gp/product/1549617214/ref=ppx_yo_dt_b_asin_title_o00_s00?ie=UTF8&psc=1)

## Free Online Book

- Interpretable Machine learning: A Guide for Making Black Box Models Explainable <https://christophm.github.io/interpretable-ml-book/>
- Elements of Statistical Learning [https://web.stanford.edu/~hastie/ElemStatLearn/printings/ESLII\\_print12.pdf](https://web.stanford.edu/~hastie/ElemStatLearn/printings/ESLII_print12.pdf)
- Hands on Machine Learning with Sklearn [https://www.amazon.com/\\_/dp/1492032646?tag=oreilly20-20](https://www.amazon.com/_/dp/1492032646?tag=oreilly20-20)  
(note: Use your GW email to login in order to be able to use the book for free)
- Neural Network Design <https://hagan.okstate.edu/NNDesign.pdf>

## MATLAB Resources:

### Videos

- Complete MATLAB Tutorial for Beginners <https://www.youtube.com/watch?v=qGiKv3-02vw>
- Understanding the Classification App <https://www.mathworks.com/videos/classify-data-using-the-classification-learner-app-106171.html>

### Free self-paced training courses

- While logged into MATLAB, MATLAB also provides the user with free self-paced training courses. Select Home tab >> Learn MATLAB.

## ACKNOWLEDGEMENTS

### Testing

Cynthia Hernandez

### Data

Publication in preparation with collaborators.

Table 1. Significance test between the top 5 important predictors for differentiating WT-Basal from SKO-Basal mice

WT-Basal vs SKO-Basal	
Top 5 Important Predictors	p-value
<i>E. coli NCTC13441</i>	3.327e-09
<i>L. gasseri DSM14869</i>	0.0009046
<i>B. zoogloeformans ATCC33285</i>	0.001944
<i>B. caccae ATCC43185</i>	0.0001295
<i>B. pseudolongum DSM20092</i>	0.02492 <sup>1</sup>

Ensemble bagged trees model had an accuracy of 78.1%

Man—Whitney significance test (WT-Basal, n=16; SKO-Basal, n=16)

<sup>1</sup>Exact p-value could not be computed due to ties (matching values within the WT-Basal dataset)

Table 2. Significance test between the top 5 important predictors for differentiating WT-Basal from WT-Tumor-PBS mice

WT-Basal vs WT-Tumor-PBS	
Top 5 Important Predictors	p-value
<i>B. caecimuris I48</i>	0.001077
<i>Halomonas sp. N32A</i>	8.158e-06
<i>B. dorei CL03T12C01</i>	9.79e-05
<i>B. vulgatus mpk</i>	0.0005384
<i>B. pseudolongum PV82</i>	0.06588 <sup>1</sup>

Ensemble bagged trees model had an accuracy of 78.3%

Man—Whitney significance test (WT-Basal, n=16; WT-Tumor-PBS, n=7)

<sup>1</sup>Exact p-value could not be computed due to ties (matching values within the WT-Basal dataset)

Table 3. Significance test between the top 5 important predictors for differentiating WT-Basal from WT-Tumor-5FU mice

WT-Basal vs WT-Tumor-5FU	
Top 5 Important Predictors	p-value

<i>E. coli</i> NCTC13441	8.158e-06
<i>A. finegoldii</i> DSM17242	0.005939
<i>L. johnsonii</i> FI9785	0.04688
<i>A. shahii</i> WAL8301	0.002676
<i>Halomonas</i> sp. N32A	3.052e-05

Ensemble bagged trees model had an accuracy of 95.7%

Man—Whitney significance test (WT-Basal, n=16; WT-Tumor-5FU, n=7)

Table 4. Significance test between the top 5 important predictors for differentiating SKO-Basal from SKO-Tumor-5FU mice

SKO-Basal vs SKO-Tumor-5FU	
Top 5 Important Predictors	p-value
<i>E. coli</i> NCTC13441	3.765e-07
<i>B. dorei</i> isolate HS1L3B079	0.000186
<i>H. hepaticus</i> ATCC51449	5.234e-05
<i>B. vulgatus</i> ATCC8482	0.135*
<i>B. caccae</i> ATCC43185	0.01223

Ensemble bagged trees model had an accuracy of 57.7%

Man—Whitney significance test (SKO-Basal, n=16; SKO-Tumor-5FU, n=10)

\*No statistical significance (p-value > 0.05)

Table 5. Significance test between the top 5 important predictors for differentiating SKO-Basal from SKO-Tumor-PBS mice

SKO-Basal vs SKO-Tumor-PBS	
Top 5 Important Predictors	p-value
<i>E. coli</i> NCTC13441	1.00*
<i>L. gasseri</i> DSM14869	0.881*
<i>B. zoogloeformans</i> ATCC33285	0.3196*
<i>B. caccae</i> ATCC43185	0.2144*
<i>B. pseudolongum</i> DSM20092	0.834*

Ensemble bagged trees model had an accuracy of 58.3%

Man—Whitney significance test (TGF-Basal, n=16; SKO-Tumor-PBS, n=8)

\*No statistical significance (p-value > 0.05)

Tables Description (this will be embedded in the text):

Classification models were build using MATLAB's Classification Application to predict mouse treatment type. After assessing the performance of the ensemble bagged trees model, the top 5 important predictors were computed. Statistical significance tests were performed using Mann-Whitney U test on the top 5 important predictors in each pairwise comparison.