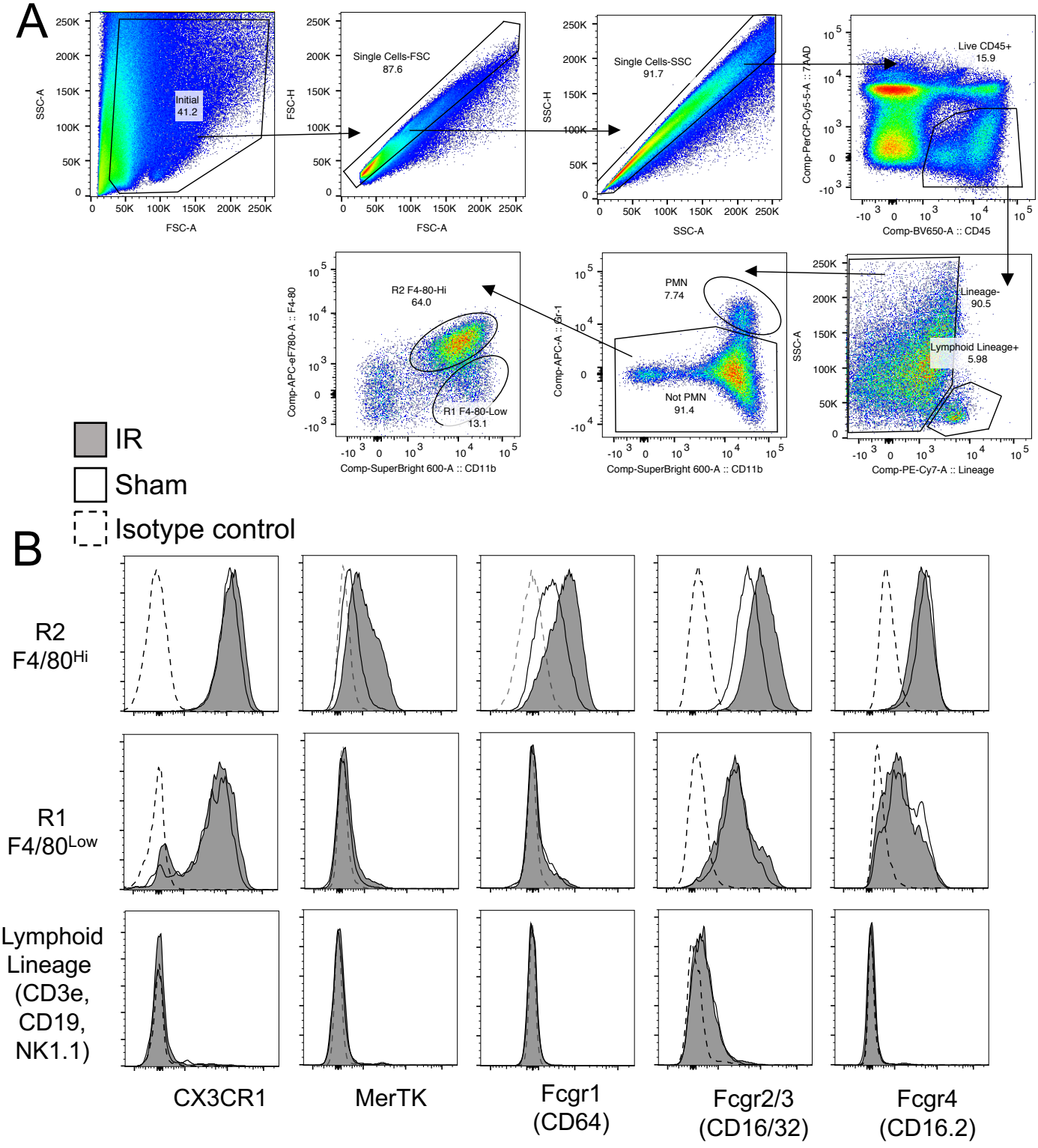
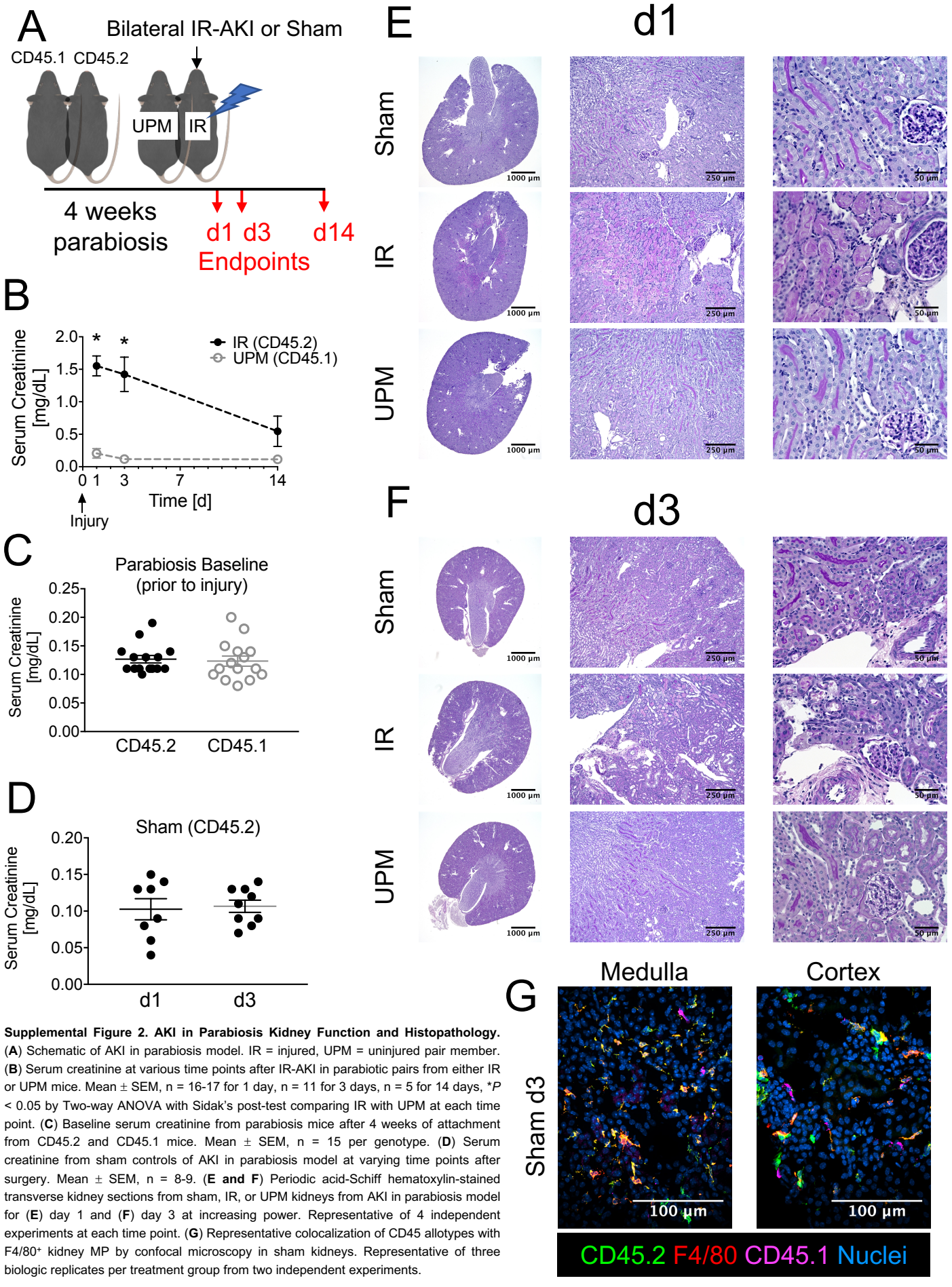
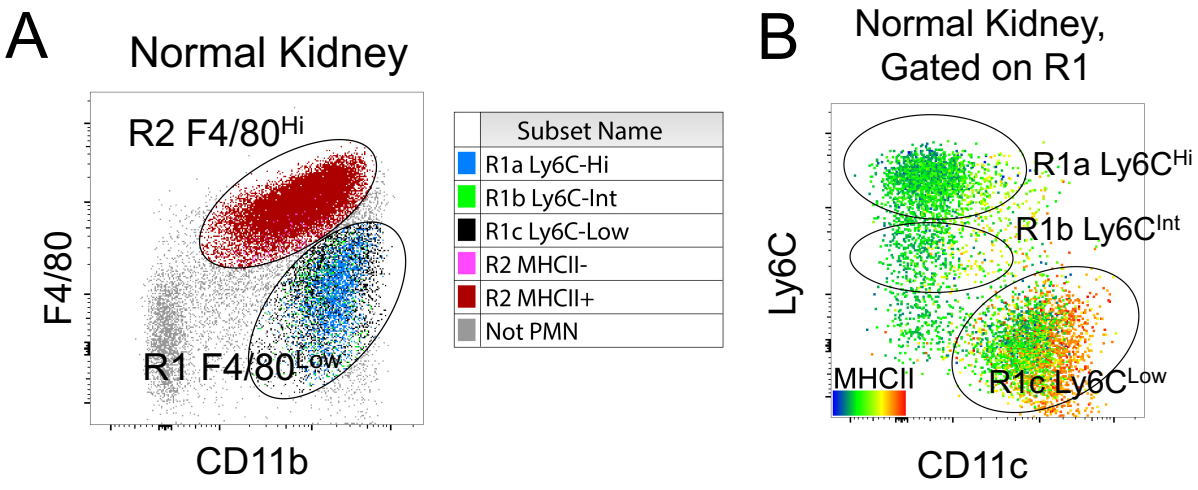
Supplemental Figures



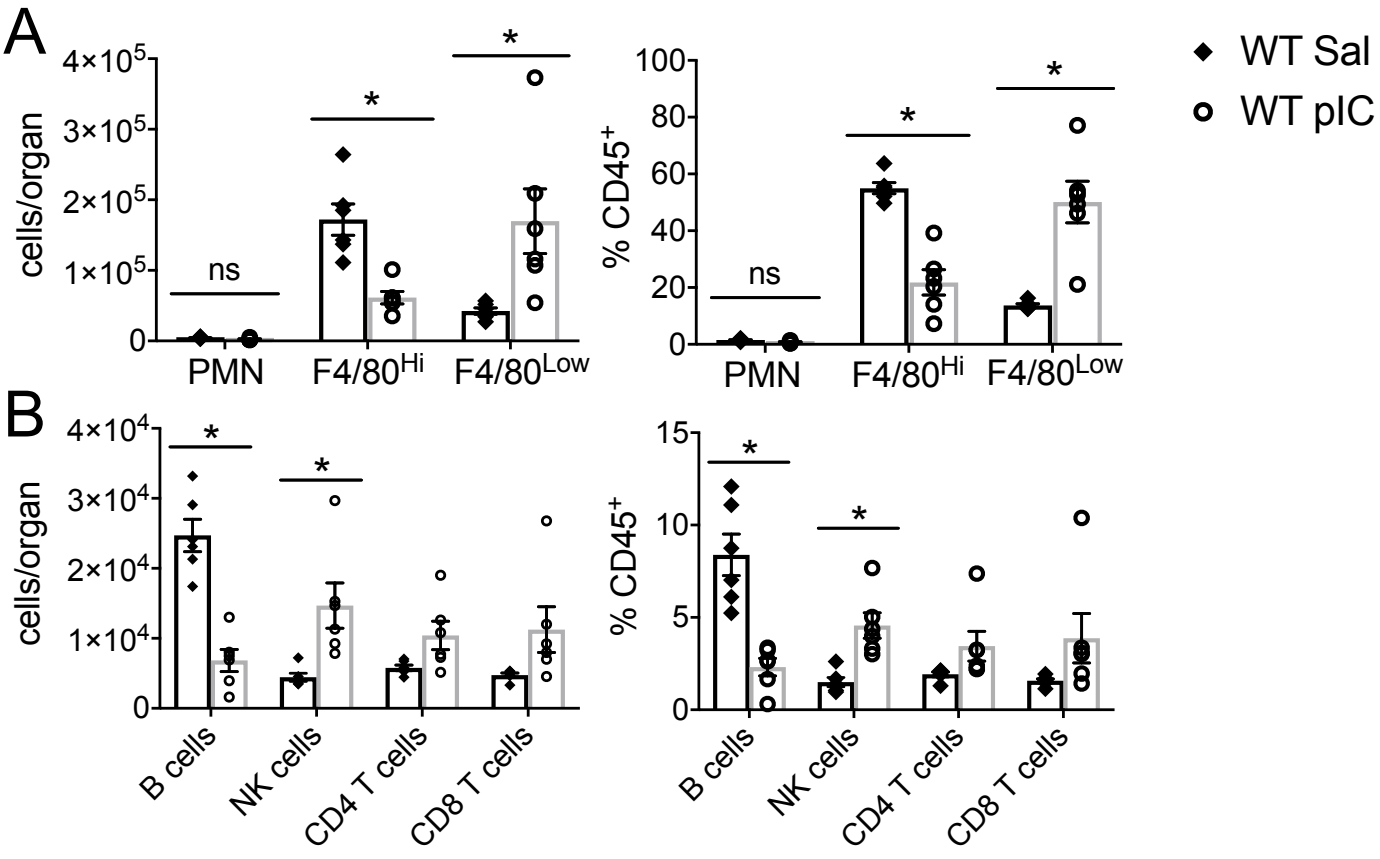
Supplemental Figure 1. Gating Strategy and Surface Phenotyping of R2 KRM and R1 MP. (**A**) Gating strategy from healthy kidney single cell suspension using pseudocolor default settings in FlowJo v10. Values are percent gated. (**B**) Single parameter flow histograms with overlays from ischemia-reperfusion (IR) and sham conditions and isotype control staining of pooled kidney samples. Representative of 1 experiment, n = 3 per treatment condition.



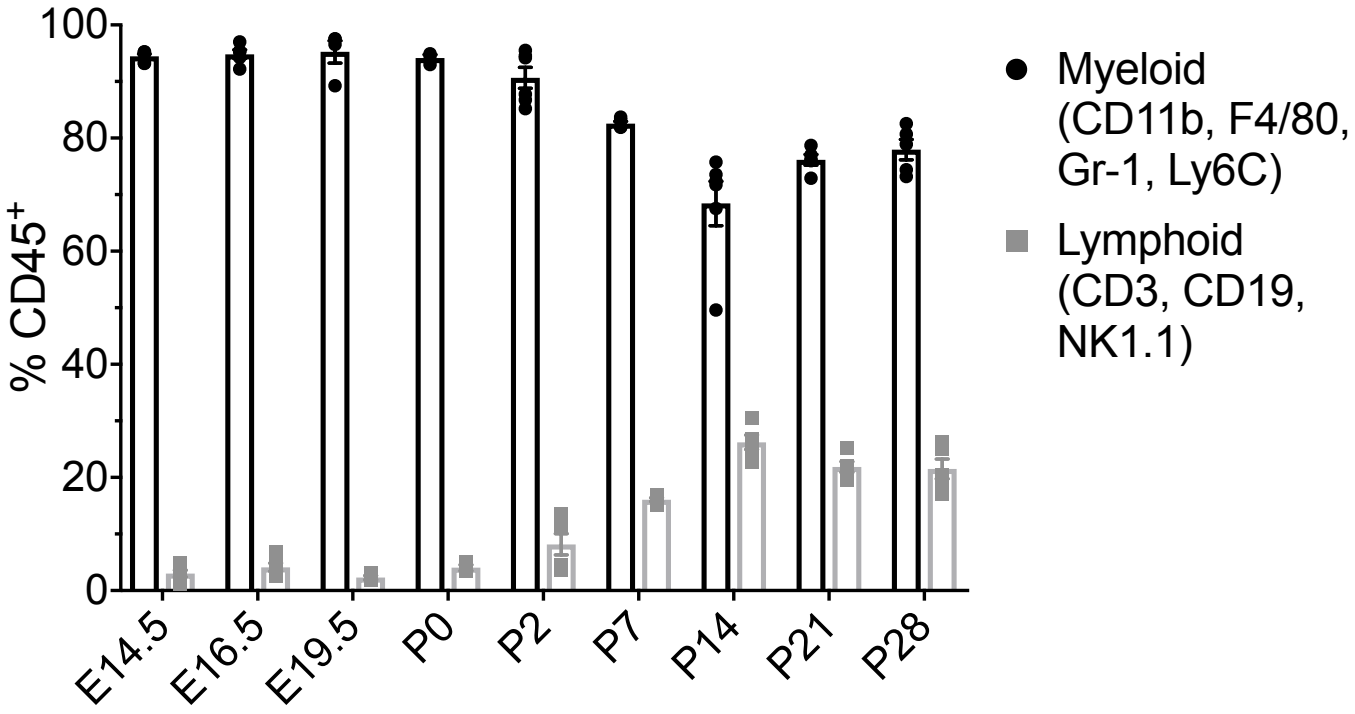
Supplemental Figure 2. AKI in Parabiosis Kidney Function and Histopathology. (**A**) Schematic of AKI in parabiosis model. IR = injured, UPM = uninjured pair member. (**B**) Serum creatinine at various time points after IR-AKI in parabiotic pairs from either IR or UPM mice. Mean ± SEM, n = 16-17 for 1 day, n = 11 for 3 days, n = 5 for 14 days, **P* < 0.05 by Two-way ANOVA with Sidak's post-test comparing IR with UPM at each time point. (**C**) Baseline serum creatinine from parabiosis mice after 4 weeks of attachment from CD45.2 and CD45.1 mice. Mean ± SEM, n = 15 per genotype. (**D**) Serum creatinine from sham controls of AKI in parabiosis model at varying time points after surgery. Mean ± SEM, n = 8-9. (**E and F**) Periodic acid-Schiff hematoxylin-stained transverse kidney sections from sham, IR, or UPM kidneys from AKI in parabiosis model for (**E**) day 1 and (**F**) day 3 at increasing power. Representative of 4 independent experiments at each time point. (**G**) Representative colocalization of CD45 allotypes with F4/80$^+$ kidney MP by confocal microscopy in sham kidneys. Representative of three biologic replicates per treatment group from two independent experiments.



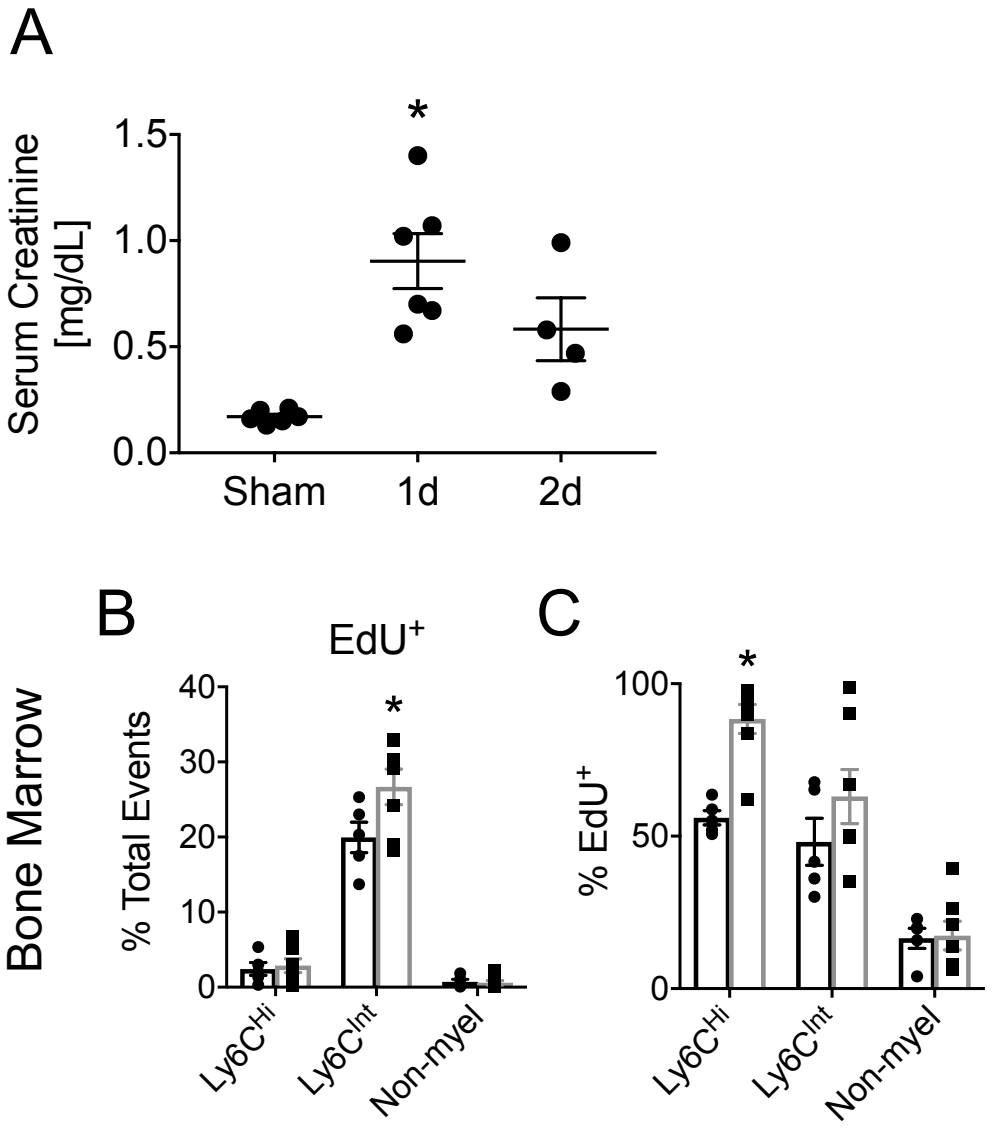
Supplemental Figure 3. Kidney Mononuclear Phagocyte (MP) Surface Phenotype Backgating and R1 Expression of Ly6C, CD11c, and MHCII Protein. 2-parameter flow histograms of $CD45^{+}$ lymphoid-lineage(CD3,CD19,NK1.1)$^{-}$ kidney MP in healthy mice with backgating color for R1a $Ly6C^{Hi}$, R1b $Ly6C^{Int}$, R1c $Ly6C^{Low}$, R2 $MHCII^{-}$, and R2 $MHCII^{+}$. R1 cells are plotted for Ly6C and CD11c expression with a FlowJo color map “median” statistic for MHCII protein. Representative of 5 independent experiments.



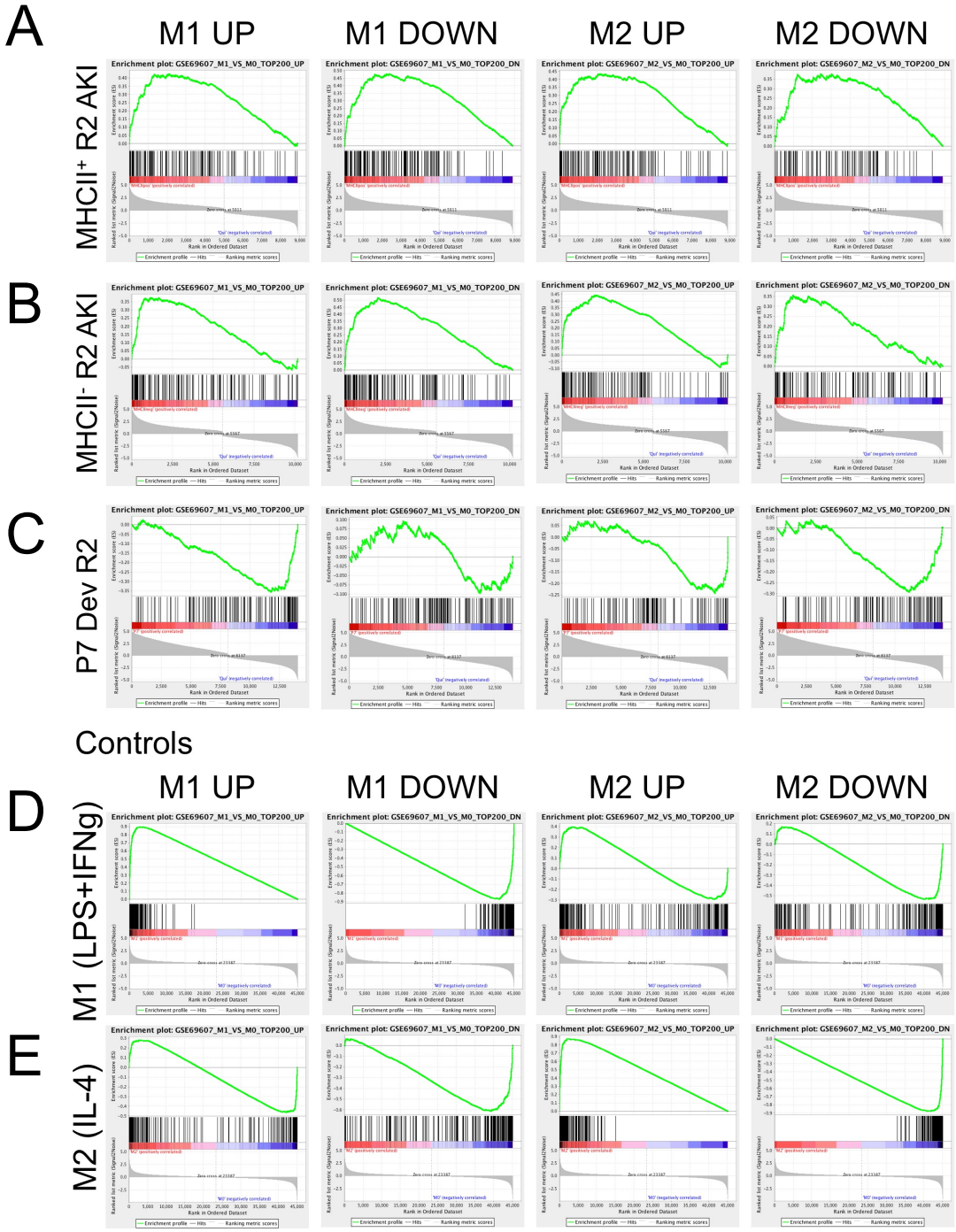
Supplemental Figure 4. Intrarenal Immune Response to Poly I:C Treatment in Wild-type C57BL/6J Adult Male Mice. (**A and B**) Flow cytometry measurements of absolute numbers (cells/organ) or proportions (% of $CD45^+$ cells) of intrarenal leukocytes after the poly I:C treatment regimen for (**A**) myeloid and (**B**) lymphoid lineage cells. Mean ± SEM, $P < 0.05$ by Two-way ANOVA with Sidak's post-test comparing saline (Sal) versus poly I:C (pIC). Data are from n = 6 biologic replicates from 1 independent experiment.



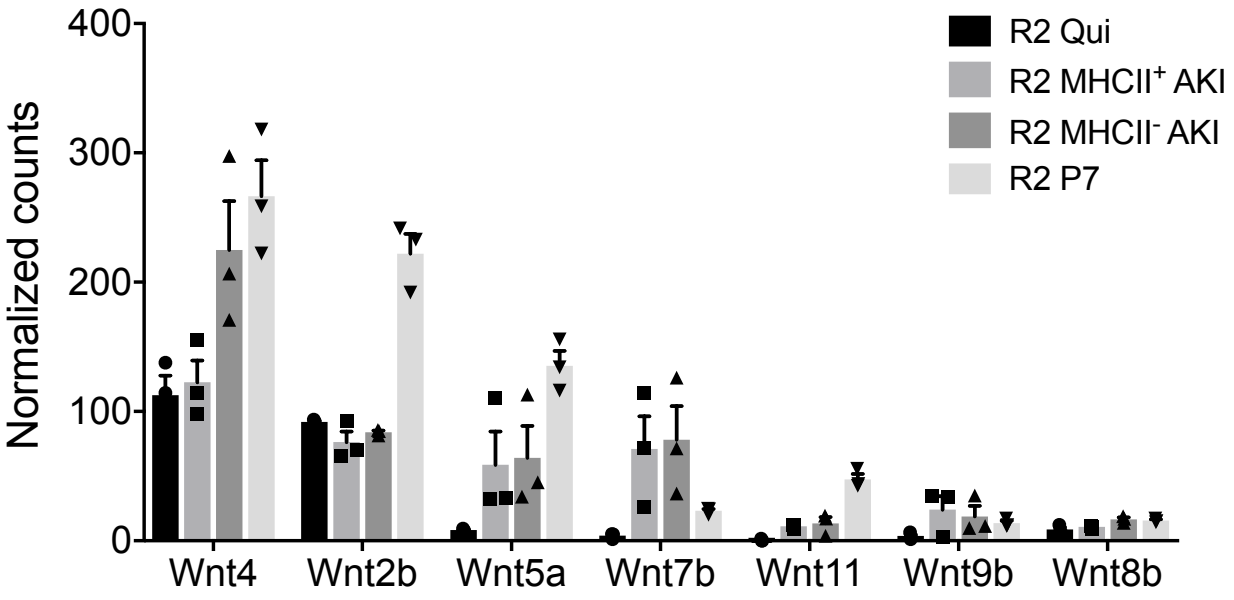
Supplemental Figure 5. Changes in Intrarenal Inflammatory Cell Lineage Predominance During Development. Proportional measurements (% of CD45$^+$ cells) of myeloid and lymphoid lineage leukocytes inside kidneys at various stages during development. E = day of embryonic life (pre-birth), P = days post-parturition (post-birth). Mean ± SEM, n = 3-6.



Supplemental Figure 6. Kidney Function and Bone Marrow Ly6C-Expressing Cell Proliferation After Ischemia-Reperfusion (IR) AKI. (**A**) Serum creatinine from sham or bilateral IR-AKI adult male C57BL/6J mice at varying time points. Mean ± SEM, n = 4-6, **P* < 0.05 by One-way ANOVA with Dunnett's post-test comparing 1d or 2d versus sham. Data are from 2 independent experiments. (**B and C**) EdU flow cytometry assay for cell cycle S-phase in bone marrow from sham or bilateral IR-AKI mice. EdU was injected intraperitoneally 3 days after injury and harvest performed at 4 days after injury. Mean ± SEM, n = 5-7, **P* < 0.05 by Two-way ANOVA with Sidak's post-test comparing IR versus sham, from 2 independent experiments. Within given subpopulations, (**B**) indicates percent total events that were EdU^+ (e.g. $Ly6C^{Int}EdU^+$/total events, pseudo-absolute number measurement) and (**C**) indicates frequency of a given subpopulation that was EdU^+ (proportional measurement). EdU = 5-ethynyl-2'-deoxyuridine, a nucleoside analog.



Supplemental Figure 7. Gene Set Enrichment Analysis (GSEA, Broad Institute) for M1/M2 Macrophage Canonical Gene Programs in KRM. (**A-E**) GSEA output plots for comparison of RNA sequencing (RNAseq) transcriptomic data from injury-responsive and P7 developmental KRM (differentially expressed genes (DEG), FDR < 0.05 vs. R2 Qui) against canonical gene sets (defined as top 100-200 DEG in M1 or M2 relative to M0 control), which define the M1 and M2 macrophage activation states in mouse. From Jablonski *et al.*, M1 and M2 activation state transcriptomic profiles were defined by performing RNAseq on untreated mouse bone marrow-derived macrophages (M0, media only baseline control) and cells treated with either lipopolysaccharide and interferon-γ (M1) or IL-4 (M2), and expression data were deposited in Gene Expression Omnibus (GSE69607). (**A-C**) Experimental comparisons of DEG from (**A**) MHCII$^+$ R2 AKI, (**B**) MHCII$^-$ R2 AKI, and (**C**) P7 Dev R2 KRM (each vs. R2 Qui, FDR < 0.05) against canonical M1/M2 activation state gene sets. (**D and E**) Positive control comparisons of M1/M2 RNAseq expression data from Jablonski *et al.* (GSE69607) against canonical gene sets defined from the same dataset. Note the high degree of association for (**D**) M1 expression data with the M1 gene sets (upregulated and downregulated) and for (**E**) M2 expression data with M2 gene sets.



Supplemental Figure 8. Normalized Counts of Wnt Ligands Among KRM. RNA sequencing normalized counts for the most highly expressed Wnt ligands in electronically sorted KRM including quiescent baseline KRM, injury-responsive MHCII^{+} KRM, injury-responsive MHCII^{-} KRM, and P7 developmental KRM. Mean ± SEM, n = 3 independent biologic replicates.

Supplemental Tables:

Supplemental Table 1. Flow cytometry antibodies

Antigen	Fluorophore	Clone	MFR
CD11b	SuperBright 600	M1/70	Thermo Fisher
CD11b	PE	M1/70	Thermo Fisher
CD11c	Brilliant Violet 785	N418	BioLegend
CD16.2 (Fcgr4)	FITC	9.00E+09	Thermo Fisher
CD16/CD32 (Fcgr2/Fcgr3)	eFluor 450	93	Thermo Fisher
CD19	Brilliant Violet 785	6D5	BioLegend
CD19	PE-Cy7	6D5	Thermo Fisher
CD3e	APC	145-2C11	Thermo Fisher
CD3e	PE-Cy7	145-2C11	Thermo Fisher
CD4	SuperBright 600	RM4-5	Thermo Fisher
CD45.1	FITC	A20	Thermo Fisher
CD45.2	Brilliant Violet 650	104	BioLegend
CD64 (Fcgr1)	PE	X54-5/7.1	BD Biosciences
CD64 (Fcgr1)	APC	X54-5/7.1	Thermo Fisher
CD8a	eFluor 450	53-6.7	Thermo Fisher
CD8a	APC-eFluor 780	53-6.7	Thermo Fisher
CX3CR1	PE	SA011F11	Thermo Fisher
F4/80	APC-eFluor 780	BM8	Thermo Fisher
F4/80	PE	BM8	Thermo Fisher

Antigen	**Fluorophore**	**Clone**	**MFR**
Gr-1	APC	1A8-Ly6G	Thermo Fisher
Ly6C	eFluor 450	HK1.4	Thermo Fisher
Ly6G/Ly6C	APC-eFluor 780	RB6-8C5	Thermo Fisher
MerTK	PE	DS5MMER	Thermo Fisher
MHCII (I-A/I-E)	APC-eFluor 780	M5/114.15.2	Thermo Fisher
MHCII (I-A/I-E)	FITC	M5/114.15.2	Thermo Fisher
MHCII (I-A/I-E)	PerCP-Cy5.5	M5/114.15.2	BD Biosciences
NK1.1	PE	PK136	Thermo Fisher
NK1.1	PE-Cy7	PK136	Thermo Fisher

Supplemental Table 2. MGI Mammalian Phenotype Ontology Analysis for Downregulated Genes from 3-Way Intersection in Venn Diagram (FDR < 0.05)

Term	Overlap (Genes in 3-way Intersection/Total Phenotype Gene Set)	FDR (*Q*-value)	Score
Decreased B Cell Proliferation	27/70	0.003	36.3
Decreased IgM Level	29/85	0.01	30.6
Abnormal B Cell Physiology	29/91	0.04	28.6

Supplemental Table 3. GSEA Analysis Output Parameters

	Gene Set	ES = enrichment score	FDR (*Q*-value)	Leading Edge
MHCII^{+} R2 AKI	M1 vs M0 UP	0.43	0.808	tags=45%, list=24%, signal=58%
	M1 vs M0 DN	0.48	0.579	tags=50%, list=25%, signal=66%
	M2 vs M0 UP	0.43	0.815	tags=45%, list=24%, signal=59%
	M2 vs M0 DN	0.38	0.461	tags=42%, list=27%, signal=57%
MHCII^{-} R2 AKI	M1 vs M0 UP	0.38	0.357	tags=28%, list=11%, signal=32%
	M1 vs M0 DN	0.52	0.252	tags=50%, list=20%, signal=61%
	M2 vs M0 UP	0.44	0.313	tags=48%, list=21%, signal=60%
	M2 vs M0 DN	0.36	0.252	tags=25%, list=11%, signal=28%
P7 Dev R2	M1 vs M0 UP	-0.36	0.860	tags=37%, list=16%, signal=44%
	M1 vs M0 DN	-0.10	1.000	tags=23%, list=20%, signal=29%
	M2 vs M0 UP	-0.24	0.462	tags=17%, list=8%, signal=19%
	M2 vs M0 DN	-0.29	0.385	tags=40%, list=20%, signal=49%
M1 Control Expression Dataset (GSE69607, LPS + IFNg)	**M1 vs M0 UP**	**0.9**	**0.149**	**tags=89%, list=8%, signal=96%**
	M1 vs M0 DN	**-0.87**	**0.188**	**tags=83%, list=9%, signal=91%**
	M2 vs M0 UP	0.4	0.112	tags=28%, list=7%, signal=30%
	M2 vs M0 DN	-0.54	0.112	tags=42%, list=11%, signal=47%
M2 Control Expression Dataset (GSE69607, IL-4)	M1 vs M0 UP	-0.47	0.144	tags=32%, list=6%, signal=34%
	M1 vs M0 DN	-0.61	0.119	tags=45%, list=13%, signal=51%
	M2 vs M0 UP	**0.87**	**0.146**	**tags=72%, list=5%, signal=75%**
	M2 vs M0 DN	**-0.88**	**0.155**	**tags=86%, list=9%, signal=94%**

Supplemental Table 4. Top Statistically Significant Enriched Pathways from Ingenuity Pathway Analysis* for DEG in 3-Way Intersection of Venn Diagram

Ingenuity Canonical Pathways	**-log(*P*-value)**	**Ratio (Percent DEG from 3-Way Intersect Relative to Total Pathway)**
[Neuronal Axon] Guidance Signaling**	14.4	27.6%
Hepatic Fibrosis / Hepatic Stellate Cell Activation	13.1	35.3%
GP6 Signaling Pathway	11.1	37.3%
Role of Osteoblasts, Osteoclasts and Chondrocytes in Rheumatoid Arthritis**	11	30.9%
Role of Macrophages, Fibroblasts and Endothelial Cells in Rheumatoid Arthritis**	9.41	26.9%
Virus Entry via Endocytic Pathways	8.42	35.3%
Leukocyte Extravasation Signaling	8.15	28.9%
Protein Kinase A Signaling	7.91	24.2%
Granulocyte Adhesion and Diapedesis	6.96	28.7%
Wnt/β-catenin Signaling	6.89	29.1%

*Takes into account direction of change relative to Qui control as well as expression levels

**Contained genes also listed in Wnt/β-catenin Signaling pathway

Supplemental Methods

Mark E. Pepin, MS

01/16/2018

Table of Contents

Code Author: Mark E. Pepin **Contact**: pepinme@uab.edu **Institution**: University of Alabama at Birmingham
Location: 542 Biomedical Research Building 2, Birmingham, AL 35294

RNA-Sequencing Analysis

RNA-Sequencing Alignment using STAR

High-throughput RNA sequencing was performed at the Heflin Genomics Core at the University of Alabama at Birmingham. Once sample read quality was checked (multiQC analysis), the paired-end fastq files were then aligned to the reference genome, which was created using Gencode M18 mouse sequence (GRCm38.p6) and annotation (gencode.vM17.chr_patch_hapl_scaff.annotation.gtf). STAR aligner is the current gold-standard for this, which we used for the current analysis. Before aligning each fastq file to

the genome, an annotated reference genome must first be assembled. This was performed as follows (this was performed in the "Cheaha" compute cluster as `bash GenomeReference.sh':

STAR=../../Tools/STAR-2.5.3a/bin/Linux_x86_64/STAR

$STAR
–runThreadN 12
–runMode genomeGenerate
–genomeDir ./
–genomeFastaFiles /data/scratch/pepinme/huHrt/Input/Genome/GRCm38.p6.genome.fa
Alignment of short reads to this annotated genome could then proceed, using the following SLURM batch script which was submitted to the UAB *Cheaha* compute cluster (See **Appendix**). This shell script contains the following STAR alignment run settings:

$STAR_RUN
–genomeDir $GENOME_DIR
–readFilesCommand zcat
–readFilesIn $INPUT_DIR/fastq/$ {VAR}_R1_001.fastq.gz
$INPUT_DIR/fastq/$ {VAR}*R2_001.fastq.gz*
–sjdbGTFfile $GENOME_DIR/gencode.vM17.chr_patch_hapl_scaff.annotation.gtf
–sjdbOverhang 99
–quantMode GeneCounts
–runThreadN 12
–outSAMtype BAM SortedByCoordinate
–outFileNamePrefix RESULTS_DIR/Alignment/{VAR}

Read Count Compiling

Before the DESeq2-based differential expression can be computed, the counts generated by STAR need to be compiled. Conveniently, the row order (ensembl gene id) is exactly the same, so we can simply merge the files together.

```
Count.files <- list.files(path = "../1_Input/1_RNA/", pattern = "*-
output_basename.counts", full.names = TRUE, all.files = TRUE)
Counts <- lapply(Count.files, read.table)
#Create a data.frame containing the raw counts
countData.raw <- as.data.frame(sapply(Counts, function(x) x[,2])) #selects
only the 2nd column as the raw counts.
#Generate Column names and Row names for the counts (remove the extra
nonsense from the path names)
colnames <- gsub( "-output_basename[.]counts", "", Count.files)
colnames <- gsub( "[.][.]/1_Input/1_RNA//", "", colnames)
colnames(countData.raw) <- colnames
row.names(countData.raw) <- Counts[[1]][,1]
countData.raw<-countData.raw[1:(nrow(countData.raw)-5),]
rownames(countData.raw)<-make.unique(gsub("\\..*", "",
rownames(countData.raw)), sep = ".")
```

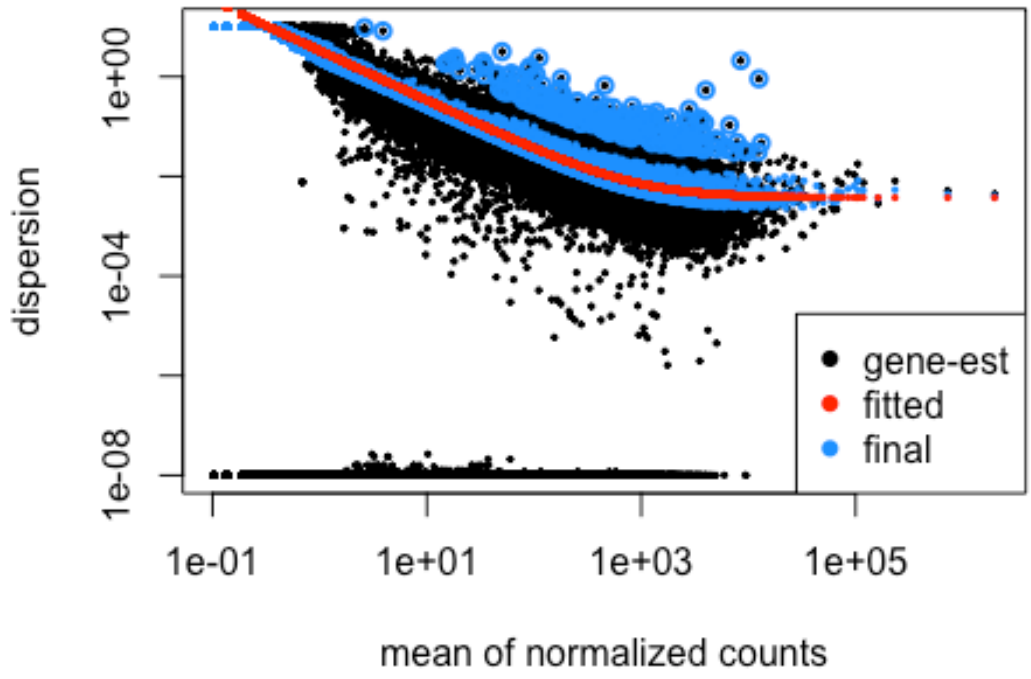
Data Pre-Processing

After alignment of the fastq files to the annotated genome assembly (mm10), the first step in the analysis is to consolidate the raw data from the provided files into data matrix that can be used to generate a normalized count matrix and differential expression dataset.

Count Normalization

DESeq2 (version 1.18.1) was used to perform the raw count normalization within R (version 3.4.2)

```
##Import count data
library(openxlsx)
library(DESeq2)
# RUN DESeq2
dds<-DESeqDataSetFromMatrix(countData=countData, colData = colData, design=
~CONDITION)
dds

## class: DESeqDataSet
## dim: 48795 6
## metadata(1): version
## assays(1): counts
## rownames(48795): ENSMUSG00000000001 ENSMUSG00000000003 ...
##   ENSMUSG00000110718 ENSMUSG00000110719
## rowData names(0):
## colnames(6): 95_R2_Qui 96_R2_Qui ... R2_MHCII_P7-2 R2_MHCII_P7-3
## colData names(5): Sample_ID MHC2 Origin CONDITION GROUP

dds$CONDITION<-relevel(dds$CONDITION, ref = "QUI")
#Determine the Dispersion Relationship (determines which distribution to use
for the differential analysis) - should take about 2 minutes
dds <- estimateSizeFactors(dds)
dds <- estimateDispersions(dds)
plotDispEsts(dds)
```




```
png(file=paste0("../2_Output/", DESCRIPTION, "_Dispersion.png"))
plotDispEsts(dds)
dev.off()

## quartz_off_screen
##                 2
```

There appears to be a linear negative correlation between the mean and dispersion estimates, so the parametric "Wald" model should be an appropriate fit for differential expression analysis. Furthermore, we could get away with the parametric fit-type, but the run-time is not significantly impaired, allowing us to use the 'local' fit-type. NOTE: If it were nonlinear throughout, we would require a 'local' nonparametric fit-type.

Differential Expression Analysis

```
##Pre-Filter to reduce the size of this dataset (according to the DESeq2
document reccomendations)
dds

## class: DESeqDataSet
## dim: 48795 6
## metadata(1): version
## assays(2): counts mu
## rownames(48795): ENSMUSG00000000001 ENSMUSG00000000003 ...
```

```
##   ENSMUSG00000110718 ENSMUSG00000110719
## rowData names(10): baseMean baseVar ... dispOutlier dispMAP
## colnames(6): 95_R2_Qui 96_R2_Qui ... R2_MHCII_P7-2 R2_MHCII_P7-3
## colData names(6): Sample_ID MHC2 ... GROUP sizeFactor

#################Run DESeq2 differential quantification (Likelihood ratio test
(LRT) or Wald-test)
dds<-DESeq(dds, test="Wald", fitType="parametric")
#compile the results tables
resdf<-as.data.frame(DESeq2::results(dds))
resdf$ensembl_gene_id<-as.factor(row.names(resdf))
```

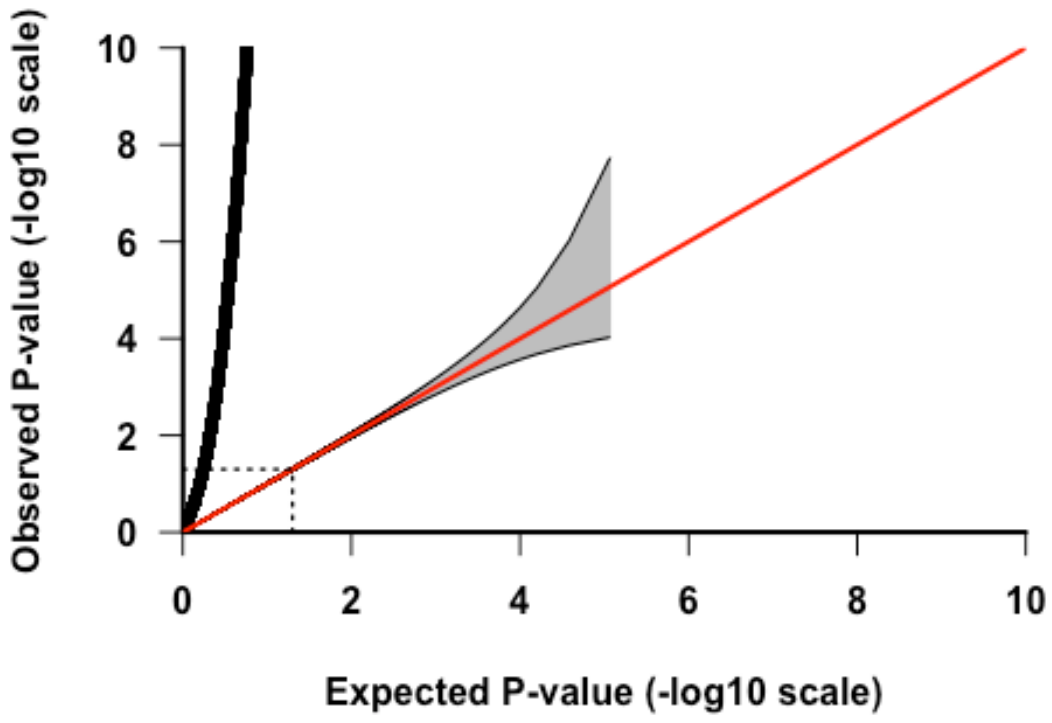
Once the differential Expression analysis was performed, the following were compiled into a results data matrix: Log2FoldChange, P-value, Bonferroni-Adjusted P-Value (Q-value), and normalized counts for each sample.

```
####Add Annotation to the results file (this will take some time, about 5
minutes...)
##Add Gene Information
library(biomaRt)
mmusculus <- useMart("ensembl", dataset="mmusculus_gene_ensembl")
bm <- getBM(attributes=c("ensembl_gene_id", "external_gene_name",
"chromosome_name", "start_position", "end_position"),  mart=mmusculus)
write.csv(bm, "../1_Input/BiomaRt_Annotation.csv")
bm<-read.csv("../1_Input/BiomaRt_Annotation.csv", row.names = 1)
bm$ensembl_gene_id<-as.character(bm$ensembl_gene_id)
resdf$ensembl_gene_id<-as.character(resdf$ensembl_gene_id)
results<-dplyr::inner_join(resdf, bm, by="ensembl_gene_id")

####Add normalized count data (for heatmap and sota)
normcount<-as.data.frame(counts(dds, normalized=TRUE))
normcount$ensembl_gene_id<-rownames(normcount)
results<-dplyr::left_join(results, normcount, by="ensembl_gene_id")
#Create filters as tabs
results_p05<-dplyr::filter(results, pvalue<0.05)
results_q05<-dplyr::filter(results, padj<0.05)

library(openxlsx)
wb_DESeq<-createWorkbook()
#Unfiltered
  addWorksheet(wb_DESeq, "Unfiltered")
  writeData(wb_DESeq, "Unfiltered", results, startCol = 1)
#P-value Significant (0.05)
  addWorksheet(wb_DESeq, "P < 0.05")
  writeData(wb_DESeq, "P < 0.05", results_p05, startCol = 1)
#Q-value Significant (0.05)
  addWorksheet(wb_DESeq, "Q < 0.05")
  writeData(wb_DESeq, "Q < 0.05", results_q05, startCol = 1)
saveWorkbook(wb_DESeq, file = paste0("../2_Output/", DESCRIPTION,
"_DESeq2.xlsx"), overwrite = TRUE)
```

QQ Plot

Before we examined the gene networks and pathways differentially regulated, the first task was to determine whether phenotypic difference exist between the two populations being compared. An effective way of determining this is the QQ plot, which compares the P-value distribution produced by the pairwise comparison (transgenic vs. WT mouse) to that of a random normal distribution. Below, it is evident that the two experimental groups produce robustly divergent expression patterns consistent with a true population difference worthy of differential expression analysis.

```
#Create Q-Q plot
test<-results
test<-test[complete.cases(test),]
pQQ(test$pvalue, lim=c(0,10))
```




```
png(file=paste0("../2_Output/", DESCRIPTION, "_QQ.Plot.png"))
pQQ(test$pvalue, lim=c(0,10))
dev.off()

## quartz_off_screen
##                 2
```

Heatmap Visualization (P < 0.01)

In order to visualize the distribution of differentially expressed genes, hierarchical clustering and heatmap visualization were performed at the Q < 0.05 statistical level. This analysis reveals that P < 0.05 is sufficient to separate all samples.

```
library(pheatmap)
results_p05<-filter(results, pvalue<0.01)
hm_data<-data.matrix(results_p05[,12:17])
rownames(hm_data)<-results_p05$external_gene_name
## Import all counts data to be used in Heatmap and PCA
Counts_ALL<-read.xlsx("../2_Output/ALL_DESeq2.xlsx", sheet = "Unfiltered",
cols = c(7,12:26))
Counts_ALL<-dplyr::select(Counts_ALL, ensembl_gene_id,
`95_R2_Qui`:`98_R2_Qui`,`69_R1c_MHCII`:`73_R1c_MHCII`, `R2_MHCII-AKI-
1`:`R2_MHCII-AKI-3`, `R2_MHCII_AKI-4`:`R2_MHCII_P7-3`)
## Merge Counts with Diff. EX
Counts_Filtered<-dplyr::semi_join(Counts_ALL, results_p05, by =
"ensembl_gene_id")
rownames(Counts_Filtered)<-Counts_Filtered$ensembl_gene_id
Counts_Filtered<-data.matrix(Counts_Filtered[,2:ncol(Counts_Filtered)])
##Index file for annotating samples
rownames(colData_all)<-colData_all$Sample_ID
Index<-dplyr::select(colData_all, MHC2, Origin, CONDITION)
Index<-as.data.frame(Index)

paletteLength <- 100
myColor <- colorRampPalette(c("dodgerblue4", "white",
"gold2"))(paletteLength)
pheatmap(Counts_Filtered,
         cluster_cols=F,
         border_color=NA,
         cluster_rows=T,
         scale = 'row',
         show_colnames = T,
         show_rownames = F,
         color = myColor,
         annotation_col = Index,
         filename=paste0("../2_Output/", DESCRIPTION,
"_VST.Heatmap.P01.pdf"))

library(dendextend)
library(colorspace)
dists <- dist(t(Counts_Filtered))
hc<-hclust(dists, method = "ward.D2")
dd<-as.dendrogram(hc)
test<-reorder(dd, Index$CONDITION)
plot(test)
```

```
pdf(paste0("../2_Output/", DESCRIPTION, "_Counts_P<0.01 P7 vs.
Quiescent_Dendrogram.pdf"))
plot(test)
dev.off()

## pdf
##   3
```

Principal Components Analysis

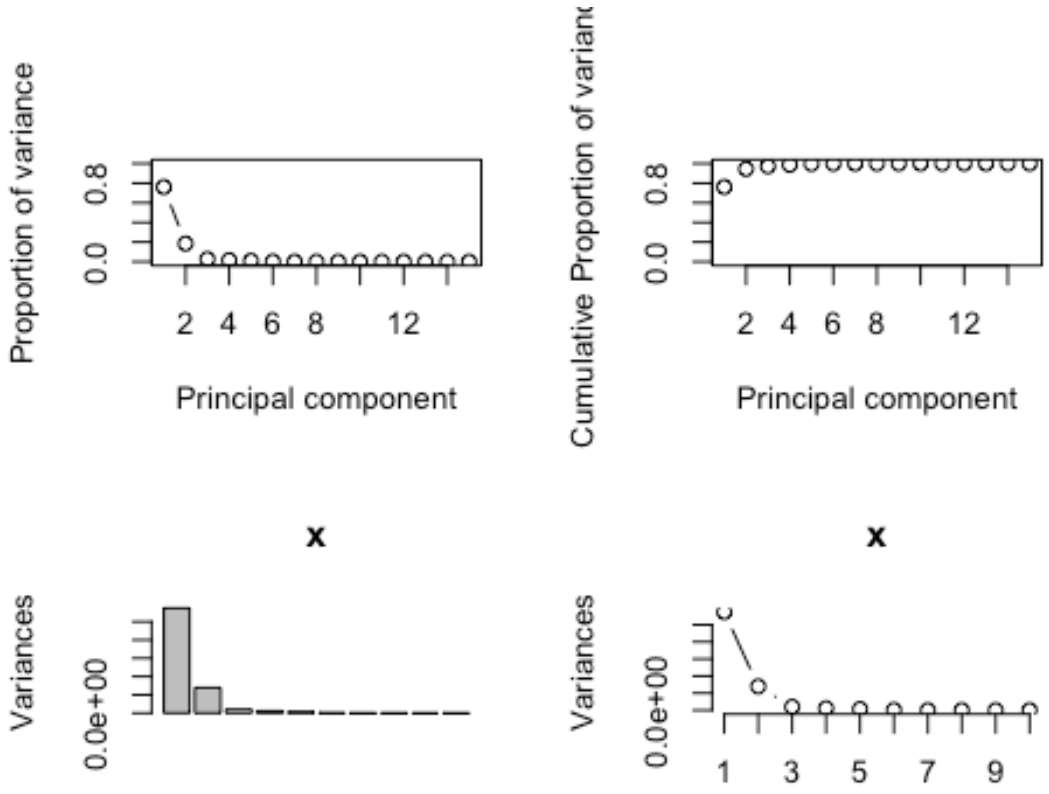
Once we established that the populations under consideration truly display divergene expression patterns, we sought to determine whether unbiased global gene expression patterns recapitulate the described phenotypes within each heart failure group. To accomplish this, an unsupervised Principal Components Analysis (PCA) was initially used with normalized counts.

PCA Features

Before running the principal components analysis, it was necessary to first determine the number of PC's required to account for 80% of the variance, a machine-learning algorithmm benchmark that provides sufficient confidence in the analysis.

```
#Plot Features of the PCA
library(readxl)
library(dplyr)
library(plotly)
##Import all count data to be used for heatmap and PCA



#transpose the dataset (required for PCA)
data.pca<-t(Counts_Filtered)
data.pca<-as.data.frame(data.pca)
##Import the data to be used for annotation
rownames(colData_all)<-colData_all$Sample_ID
Index<-colData_all
Index<-as.data.frame(Index)
##merge the file
data.pca_Final<-merge(Index, data.pca, by=0)
rownames(data.pca_Final)<-data.pca_Final$Row.names
pca.comp<-prcomp(data.pca_Final[,(ncol(Index)+2):ncol(data.pca_Final)])

pcaCharts=function(x) {
    x.var <- x$sdev ^ 2
    x.pvar <- x.var/sum(x.var)
    par(mfrow=c(2,2))
    plot(x.pvar,xlab="Principal component",
         ylab="Proportion of variance", ylim=c(0,1), type='b')
    plot(cumsum(x.pvar),xlab="Principal component",
         ylab="Cumulative Proportion of variance",
         ylim=c(0,1),
```

```
          type='b')
    screeplot(x)
    screeplot(x,type="l")
    par(mfrow=c(1,1))
}
pcaCharts(pca.comp)
```




```
png(file=paste0("../2_Output/", DESCRIPTION, "_PCA.Charts.png"))
pcaCharts(pca.comp)
dev.off()

## quartz_off_screen
##                 2
```

3-Dimensional PCA

From the previous calculations, it is seens that only 2 principal components are necessary (accounting for >80% cumulative variance). Nonetheless, below is a 3-D PCA to ensure that all groups are characterize to higher-degree of stringency.

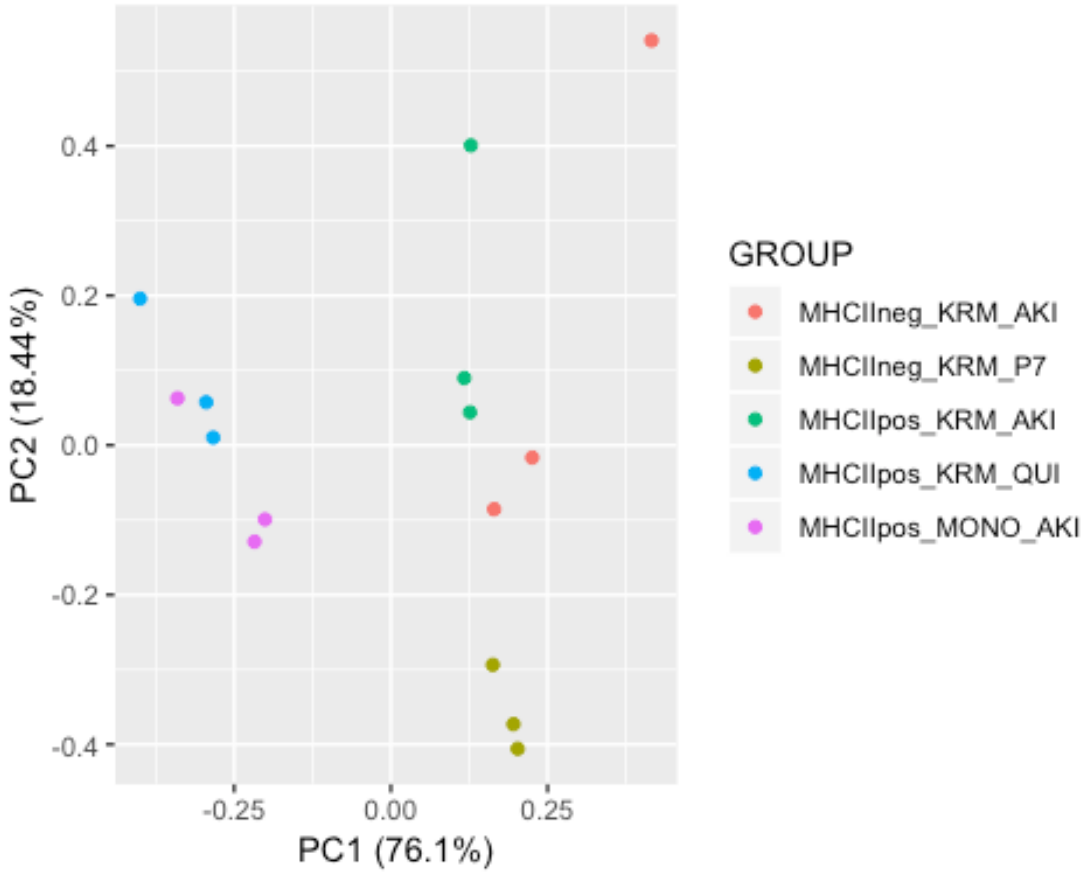
```
##Create a 3D-PCA for Inspection
library(plotly)
##Index
Index_PCA<-read.csv("../1_Input/colData_PCA.csv")
```

```r
Index_PCA$CONDITION<-as.numeric(as.character(Index_PCA$CONDITION))
Index_PCA$GROUP<-as.numeric(as.character(Index_PCA$GROUP))
rownames(Index_PCA)<-Index_PCA$Sample_ID
PCs<-merge(pca.comp$x, Index_PCA, by=0)
rownames(PCs)<-PCs$Row.names
ax_text<-list(
  family = "times",
  size = 12,
  color = "black")
t <- list(
  family = "times",
  size = 14,
  color = "black")
p <- plot_ly(PCs, x = ~PC1, y = ~PC2, z = ~PC3,
   marker = list(color = ~GROUP,
                 colorscale = c('#FFE1A1', '#683531'),
                 showscale = TRUE),
   text=rownames(PCs)) %>%
  add_markers() %>%
  add_text(textfont = t, textposition="bottom") %>%
  layout(scene = list(
     xaxis = list(title = 'PC1', zerolinewidth = 4,
        zerolinecolor="darkgrey", linecolor="darkgrey",
        linewidth=4, titlefont=t, tickfont=ax_text),
     yaxis = list(title = 'PC2', zerolinewidth = 4,
        zerolinecolor="darkgrey", linecolor="darkgrey",
        linewidth=4, titlefont=t, tickfont=ax_text),
    zaxis = list(title = 'PC3', zerolinewidth = 4,
        zerolinecolor="darkgrey",  linecolor="darkgrey",
        linewidth=4, titlefont=t, tickfont=ax_text)),
  annotations = list(
           x = 1.13,
           y = 1.03,
           text = 'Diabetes',
           xref = '1',
           yref = '0',
           showarrow = FALSE))
# p #must comment out for PDF generation via knitr (Pandoc)

library(rgl)
colors=c("black", "green", "red", "blue", "yellow")
plot3d(x=PCs$PC1, y=PCs$PC2, z=PCs$PC3, xlab="PC1", ylab="PC2", zlab="PC3",
cols=as.numeric(PCs$GROUP), size =3, type="s")

for (i in 1:360){
  rgl.viewpoint(theta=i, phi=0)
  rgl.snapshot(filename = paste0("animated/", paste("3D.Network", formatC(i,
width=3, flag="0"), sep="_"), ".png"), fmt = "png", top = TRUE )
}
```

```
system("convert -delay 5 animated/*.png 3D.Network_plot.gif")
file.remove(list.files(path="./animated/", pattern=".png"))

## logical(0)
```

2-D PCA

```
library(ggfortify)
library(cluster)
autoplot(pca.comp, data = data.pca_Final, colour = "GROUP")
```




```
pdf(file=paste0("../2_Output/", DESCRIPTION, "_2D-PCA.pdf"))
autoplot(pca.comp, data = data.pca_Final, colour = "GROUP")
dev.off()

## quartz_off_screen
##                 2
```

Venn Diagram Overlap of Differential Gene Expression

The following was used to determine the overlapping DEGs based on the 3 comparisons: P7 vs. Qui, MHCII-AKI vs. Qui, and MHCII+AKI vs. Qui.

```
library(dplyr)
library(openxlsx)
```

```r
#Import the 3 DEG Datasets to be merged
R2_MHCIIpos_AKI.vs.R2_Qui<-
read.xlsx("../2_Output/R2_MHCII_AKI.vs.R2_Qui/R2_MHCII_AKI.vs.R2_Qui_DESeq2.x
lsx",
                                    sheet = "P < 0.05", colNames = T)
rownames(R2_MHCIIpos_AKI.vs.R2_Qui)<-
R2_MHCIIpos_AKI.vs.R2_Qui$ensembl_gene_id
colnames(R2_MHCIIpos_AKI.vs.R2_Qui)[2]<-paste0("log2FoldChange", "_MHCIIpos")
colnames(R2_MHCIIpos_AKI.vs.R2_Qui)[3]<-paste0("lfcSE", "_MHCIIpos")
colnames(R2_MHCIIpos_AKI.vs.R2_Qui)[5]<-paste0("pvalue", "_MHCIIpos")
colnames(R2_MHCIIpos_AKI.vs.R2_Qui)[6]<-paste0("padj", "_MHCIIpos")
R2_MHCIIpos_AKI.vs.R2_Qui<-dplyr::select(R2_MHCIIpos_AKI.vs.R2_Qui,
contains("MHCIIpos"), external_gene_name)


R2_MHCIIpos_P7.vs.R2_Qui<-
read.xlsx("../2_Output/R2_MHCII_P7.vs.R2_Qui/R2_MHCII_P7.vs.R2_Qui_DESeq2.xls
x",
                                    sheet = "P < 0.05", colNames = T)
rownames(R2_MHCIIpos_P7.vs.R2_Qui)<-R2_MHCIIpos_P7.vs.R2_Qui$ensembl_gene_id
colnames(R2_MHCIIpos_P7.vs.R2_Qui)[2]<-paste0("log2FoldChange", "_P7")
colnames(R2_MHCIIpos_P7.vs.R2_Qui)[3]<-paste0("lfcSE", "_P7")
colnames(R2_MHCIIpos_P7.vs.R2_Qui)[5]<-paste0("pvalue", "_P7")
colnames(R2_MHCIIpos_P7.vs.R2_Qui)[6]<-paste0("padj", "_P7")
R2_MHCIIpos_P7.vs.R2_Qui<-dplyr::select(R2_MHCIIpos_P7.vs.R2_Qui,
contains("_P7"), -contains("R2_MHCII"), external_gene_name)

R2_MHCIInegAKI.vs.R2_Qui<-read.xlsx("../2_Output/R2_MHCII-
AKI.vs.R2_Qui/R2_MHCII-AKI.vs.R2_Qui_DESeq2.xlsx",
                                    sheet = "P < 0.05", colNames = T)
rownames(R2_MHCIInegAKI.vs.R2_Qui)<-R2_MHCIInegAKI.vs.R2_Qui$ensembl_gene_id
colnames(R2_MHCIInegAKI.vs.R2_Qui)[2]<-paste0("log2FoldChange", "_MHCIIneg")
colnames(R2_MHCIInegAKI.vs.R2_Qui)[3]<-paste0("lfcSE", "_MHCIIneg")
colnames(R2_MHCIInegAKI.vs.R2_Qui)[5]<-paste0("pvalue", "_MHCIIneg")
colnames(R2_MHCIInegAKI.vs.R2_Qui)[6]<-paste0("padj", "_MHCIIneg")
R2_MHCIInegAKI.vs.R2_Qui<-dplyr::select(R2_MHCIInegAKI.vs.R2_Qui,
contains("_MHCIIneg"), external_gene_name)

#Intersect the MHCIIneg_AKI and P7 relative to Quiescent
MHCIIneg.v.P7<-merge(R2_MHCIInegAKI.vs.R2_Qui, R2_MHCIIpos_P7.vs.R2_Qui,
by=0)
MHCIIneg.v.MHCIIpos<-merge(R2_MHCIInegAKI.vs.R2_Qui,
R2_MHCIIpos_AKI.vs.R2_Qui, by=0)
P7.v.MHCIIpos<-merge(R2_MHCIIpos_P7.vs.R2_Qui, R2_MHCIIpos_AKI.vs.R2_Qui,
by=0)
#Identify the 3-way intersection (center of the Venn Diagram)
three.way<-merge(MHCIIneg.v.P7, MHCIIneg.v.MHCIIpos, by="Row.names")
three.way<-dplyr::inner_join(three.way, P7.v.MHCIIpos, by="Row.names")
```

```
#subtract 3way from all intersections
MHCIIneg.v.P7_ONLY<-dplyr::anti_join(MHCIIneg.v.P7, three.way, by =
"Row.names")
MHCIIneg.v.MHCIIpos_ONLY<-dplyr::anti_join(MHCIIneg.v.MHCIIpos, three.way,
by="Row.names")
P7.v.MHCIIpos_ONLY<-dplyr::anti_join(P7.v.MHCIIpos, three.way,
by="Row.names")

##Create an IPA Upload dataset
IPA<-dplyr::full_join(R2_MHCIIpos_AKI.vs.R2_Qui, R2_MHCIInegAKI.vs.R2_Qui,
by="external_gene_name")
IPA<-dplyr::full_join(IPA, R2_MHCIIpos_P7.vs.R2_Qui, by="external_gene_name")
write.csv(IPA, "IPA_Import_P<0.05.csv")

#Save a copy of the countData
library(openxlsx)
wb_countData<-createWorkbook()

addWorksheet(wb_countData, "MHCIIneg.v.P7_ONLY")
  writeData(wb_countData, "MHCIIneg.v.P7_ONLY", MHCIIneg.v.P7_ONLY)
addWorksheet(wb_countData, "MHCIIneg.v.MHCIIpos_ONLY")
  writeData(wb_countData, "MHCIIneg.v.MHCIIpos_ONLY",
MHCIIneg.v.MHCIIpos_ONLY)
addWorksheet(wb_countData, "P7.v.MHCIIpos_ONLY")
  writeData(wb_countData, "P7.v.MHCIIpos_ONLY", P7.v.MHCIIpos_ONLY)
 addWorksheet(wb_countData, "three.way")
  writeData(wb_countData, "three.way", three.way)

saveWorkbook(wb_countData, file =
paste0("../2_Output/Venn.Diagram_Intersections.xlsx"), overwrite = TRUE)
```

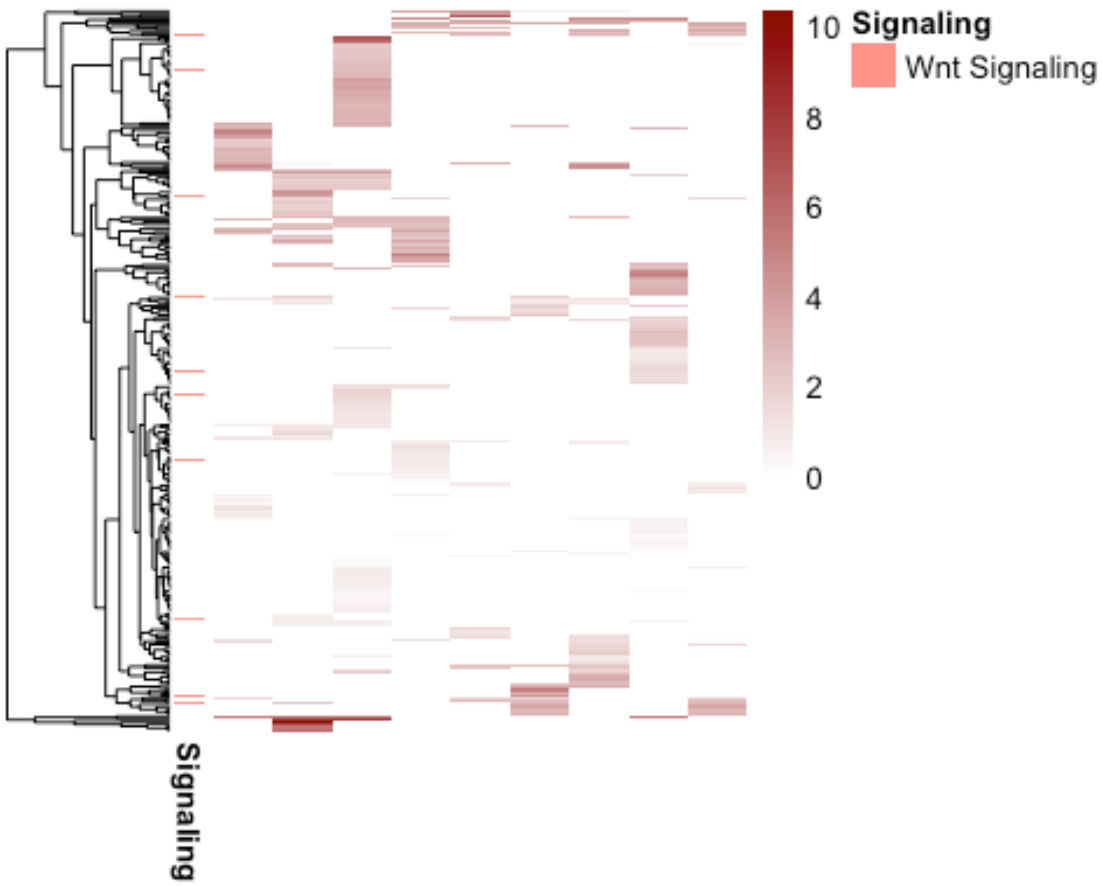
Hierarchical Clustering of MGI Pathway Enrichment Analysis

Because genes are often assigned to numerous ontology terms and curated pathways, it is useful to determine whether a subset of differentially-expressed genes is driving the enrichment of top GO-term pathways. DEGs were extracted and compiled which were responsible for driving enrichment for each of the top-10 pathways, clustered via Ward.D2, and visualized via heatmap.

```
library(dplyr)
library(openxlsx)
library(pheatmap)
library(RColorBrewer)
#Merge pathways with differential gene expression of MHCneg AKI
DEGs_MHCneg.AKI<-read.xlsx("../2_Output/R2_MHCII-AKI.vs.R2_Qui/R2_MHCII-
AKI.vs.R2_Qui_DESeq2.xlsx", sheet = "P < 0.05")
DEGs_MHCneg.AKI$external_gene_name<-
toupper(DEGs_MHCneg.AKI$external_gene_name)
ALL.Pathways<-
```

```
read.xlsx("../1_Input/Pathway.Enrichment/Enrichr_Ontology_3wayIntersect_ALL-
UP_MGI_Mammalian_Phenotype_2017_table.xlsx", sheet = "Gene.Table")
#Import the Wnt signaling genes
Wnt<-read.xlsx("../1_Input/Pathway.Enrichment/wnt.genes.xlsx")
rownames(Wnt)<-Wnt$Gene.Symbol
Wnt<-merge(Wnt, DEGs_MHCneg.AKI, by.x = "Gene.Symbol", by.y =
"external_gene_name")
rownames(Wnt)<-Wnt$Gene.Symbol
Wnt<-dplyr::select(Wnt, Signaling)
#Create a Table for each column in the Pathways dataset
one<-ALL.Pathways[,1]
one<-as.data.frame(one)
one<-merge(one, DEGs_MHCneg.AKI, by.x = "one", by.y="external_gene_name")
one<-dplyr::select(one, Gene.Symbol="one", log2FC_one="log2FoldChange")

two<-ALL.Pathways[,2]
two<-as.data.frame(two)
two<-merge(two, DEGs_MHCneg.AKI, by.x = "two", by.y="external_gene_name")
two<-dplyr::select(two, Gene.Symbol="two", log2FC_two="log2FoldChange")

three<-ALL.Pathways[,3]
three<-as.data.frame(three)
three<-merge(three, DEGs_MHCneg.AKI, by.x = "three",
by.y="external_gene_name")
three<-dplyr::select(three, Gene.Symbol="three",
log2FC_three="log2FoldChange")

four<-ALL.Pathways[,4]
four<-as.data.frame(four)
four<-merge(four, DEGs_MHCneg.AKI, by.x = "four", by.y="external_gene_name")
four<-dplyr::select(four, Gene.Symbol="four", log2FC_four="log2FoldChange")

five<-ALL.Pathways[,5]
five<-as.data.frame(five)
five<-merge(five, DEGs_MHCneg.AKI, by.x = "five", by.y="external_gene_name")
five<-dplyr::select(five, Gene.Symbol="five", log2FC_five="log2FoldChange")

# six<-ALL.Pathways[,6]
# six<-as.data.frame(six)
# six<-merge(six, DEGs_MHCneg.AKI, by.x = "six", by.y="external_gene_name")
# six<-dplyr::select(six, Gene.Symbol="six", log2FC_six="log2FoldChange")

seven<-ALL.Pathways[,7]
seven<-as.data.frame(seven)
seven<-merge(seven, DEGs_MHCneg.AKI, by.x = "seven",
by.y="external_gene_name")
seven<-dplyr::select(seven, Gene.Symbol="seven",
log2FC_seven="log2FoldChange")
```

```
eight<-ALL.Pathways[,8]
eight<-as.data.frame(eight)
eight<-merge(eight, DEGs_MHCneg.AKI, by.x = "eight",
by.y="external_gene_name")
eight<-dplyr::select(eight, Gene.Symbol="eight",
log2FC_eight="log2FoldChange")

nine<-ALL.Pathways[,9]
nine<-as.data.frame(nine)
nine<-merge(nine, DEGs_MHCneg.AKI, by.x = "nine", by.y="external_gene_name")
nine<-dplyr::select(nine, Gene.Symbol="nine", log2FC_nine="log2FoldChange")

ten<-ALL.Pathways[,10]
ten<-as.data.frame(ten)
ten<-merge(ten, DEGs_MHCneg.AKI, by.x = "ten", by.y="external_gene_name")
ten<-dplyr::select(ten, Gene.Symbol="ten", log2FC_ten="log2FoldChange")

##Combine all of the datasets
total<-full_join(one, two, by='Gene.Symbol')
total<-full_join(total, three, by='Gene.Symbol')
total<-full_join(total, four, by='Gene.Symbol')
total<-full_join(total, five, by='Gene.Symbol')
# total<-full_join(total, six, by='Gene.Symbol')
total<-full_join(total, seven, by='Gene.Symbol')
total<-full_join(total, eight, by='Gene.Symbol')
total<-full_join(total, nine, by='Gene.Symbol')
total<-full_join(total, ten, by='Gene.Symbol')

total<-distinct(total)
rownames(total)<-total$Gene.Symbol
cluster<-dplyr::select(total, -Gene.Symbol)
cluster<-data.matrix(cluster)
cluster[is.na(cluster)]<-0
rownames(cluster)<-total$Gene.Symbol

##row names to display
row.display<-one
test<-row.display$Gene.Symbol

paletteLength <- 100
myColor <- colorRampPalette(c("white", "darkred"))(paletteLength)
pheatmap(cluster, color = myColor, border_color = NA, annotation_row = Wnt,
show_rownames = F, show_colnames = F, cluster_cols = F)
```



Supplemental Table: R Session Information

All packages and setting are acquired using the following command:

```
library(kableExtra)
sinfo<-devtools::session_info()
sinfo$platform

##  setting  value
##  version  R version 3.5.1 (2018-07-02)
##  os       macOS  10.14.2
##  system   x86_64, darwin15.6.0
##  ui       X11
##  language (EN)
##  collate  en_US.UTF-8
##  ctype    en_US.UTF-8
##  tz       America/Chicago
##  date     2019-01-16

sinfo$packages %>% kable(
                          align="c",
                          longtable=T,
```

```
                        booktabs=T,
                        caption="Packages and Required Dependencies") %>%
  kable_styling(latex_options=c("striped", "repeat_header", "condensed"))
```