Contents

1 GWAS

```
library(tidyverse)
library(ggrepel)
library(patchwork)

set.seed(42)

#### Data utils

read_sheet <- function(name) {
  fs::path("..", "data", "supplementary_tables.xlsx") %>% readxl::read_excel(name)
}

# Load results from Zhou et al, NatGen 201
zhou_results <- function() {
  data <- read_sheet("S5_LDL_TWAS") %>%
    # Drop unused cols
    select(!genetype) %>%
    # Make table tidy
    mutate(chr = as.character(chr)) %>%
    mutate(across(effect_size_PrediXcan:PFDR_JTI, as.numeric)) %>%
    pivot_longer(
      !c(genename, geneid, chr, left, right),
      names_to = c(".value", "method"),
      names_pattern = "(.*)_(PrediXcan|JTI)"
    ) %>%
    # arrange by method
    arrange(desc(method)) %>%
    rename(pfdr = PFDR) %>%
    mutate(bp = left) %>%
    drop_na()
}

# Load local GWAS result
read_result <- function(ukbb_id, method, tissue) {
  fs::path("..", "results", paste0(ukbb_id, "-", method, "_", tissue, ".csv")) %>%
    read_csv() %>%
```

```
    # Match columns to zhou results
    rename(geneid = gene, genename = gene_name) %>%
    # Add new cols
    mutate(method = method, pfdr = p.adjust(pvalue, method = "fdr"))
}

# Merge results from multiple methods
read_results <- function(ukbb_id, tissue, methods = c("PrediXcan", "JTI", "UTMOST")) {
  methods %>%
    map(function(m) read_result(ukbb_id, m, tissue)) %>%
    bind_rows()
}


#### Plotting utils

COLORS <- c("#B4D88B", "#A6CEE2", "#34A048", "#1F78B4")
FDR <- 0.05

manhattan_base <- function(res) {
  # Adapted from https://www.r-graph-gallery.com/101_Manhattan_plot.html
  # get cumulative position
  res <- res %>%
    group_by(chr) %>%
    summarise(chr_len = max(bp)) %>%
    arrange(as.numeric(chr)) %>%
    mutate(tot = cumsum(chr_len) - chr_len) %>%
    select(-chr_len) %>%
    left_join(res, ., by = "chr") %>%
    mutate(bp_cum = tot + bp)

  axisdf <- res %>%
    group_by(chr) %>%
    summarize(center = (max(bp_cum) + min(bp_cum)) / 2) %>%
    arrange(as.numeric(chr))

  ggplot(res, aes(x = bp_cum, y = -log10(pvalue))) +
    scale_x_continuous(
      label = c(1:15, "", 17, "", 19, "", 21, ""),
      breaks = axisdf$center
    ) +
    scale_y_log10(
      breaks = c(1, 2, 3, 5, 10, 20, 30, 50, 100, 200, 300, 500),
      limits = c(0.7, 600),
      expand = c(0.04, 0)
    ) +
    theme_bw() +
    theme(
      panel.border = element_blank(),
      panel.grid = element_blank(),
      legend.position = "top"
    ) +
    labs(x = "Chromosome", y = "-log10[P]", color = "")
```

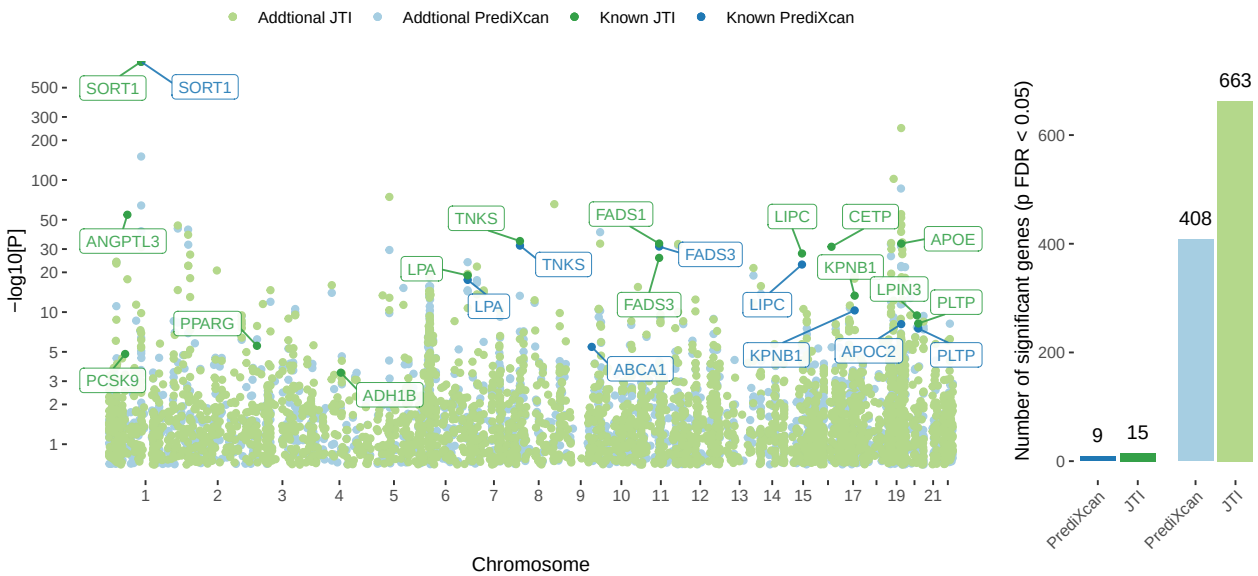
```
}

sig_genes_bars <- function(results, fdr = FDR, colors = COLORS) {
  results %>%
    filter(pfdr < fdr) %>%
    ggplot(aes(fct_rev(method), fill = paste(known, method))) +
    geom_bar(stat = "count") +
    geom_text(stat = "count", aes(label = ..count..), vjust = -1) +
    scale_y_continuous(limits = c(0, 700)) +
    facet_wrap(~ !known) +
    ylab("Number of significant genes (p FDR < 0.05)") +
    theme_bw() +
    theme(
      strip.text.x = element_blank(),
      panel.border = element_blank(),
      panel.grid = element_blank(),
      axis.title.x = element_blank(),
      axis.text.x = element_text(angle = 45, hjust = 1)
    ) +
    guides(fill = FALSE) +
    scale_fill_manual(values = colors)
}

manhatten_labeled_fdr <- function(results, fdr = FDR, colors = COLORS, gray_below = FALSE) {
  if (gray_below) {
    below_pts <- geom_point(data = ~ filter(.x, pfdr > fdr), color = "lightgray")
  } else {
    below_pts <- geom_point(data = ~ filter(.x, pfdr > fdr), aes(color = paste("Addtional", method)))
  }
  manhattan_base(results) +
    geom_point(data = ~ filter(.x, pfdr < fdr & !known), aes(color = paste("Addtional", method))) +
    geom_point(data = ~ filter(.x, pfdr < fdr & known), aes(color = paste("Known", method))) +
    below_pts +
    geom_label_repel(
      data = ~ filter(.x, pfdr < fdr & known),
      aes(label = genename, color = paste("Known", method)),
      size = 3,
      alpha = 0.9,
      show.legend = FALSE,
      box.padding = 0.5
    ) +
    scale_color_manual(values = colors)
}

# Creates barplot from paper
fig4 <- function(results, title = NULL, colors = COLORS, gray_below = FALSE) {
  if (!is.null(title)) title <- ggtitle(title)
  m <- manhatten_labeled_fdr(results, colors = colors, gray_below = gray_below) + title
  bars <- sig_genes_bars(results, colors = colors)
  m + bars + plot_layout(widths = c(5, 1))
}
```

1.1 Manhattan plot for LDL-C level from the application of PrediXcan and JTI models in liver to UK Biobank LDL-C GWAS summary statistics.

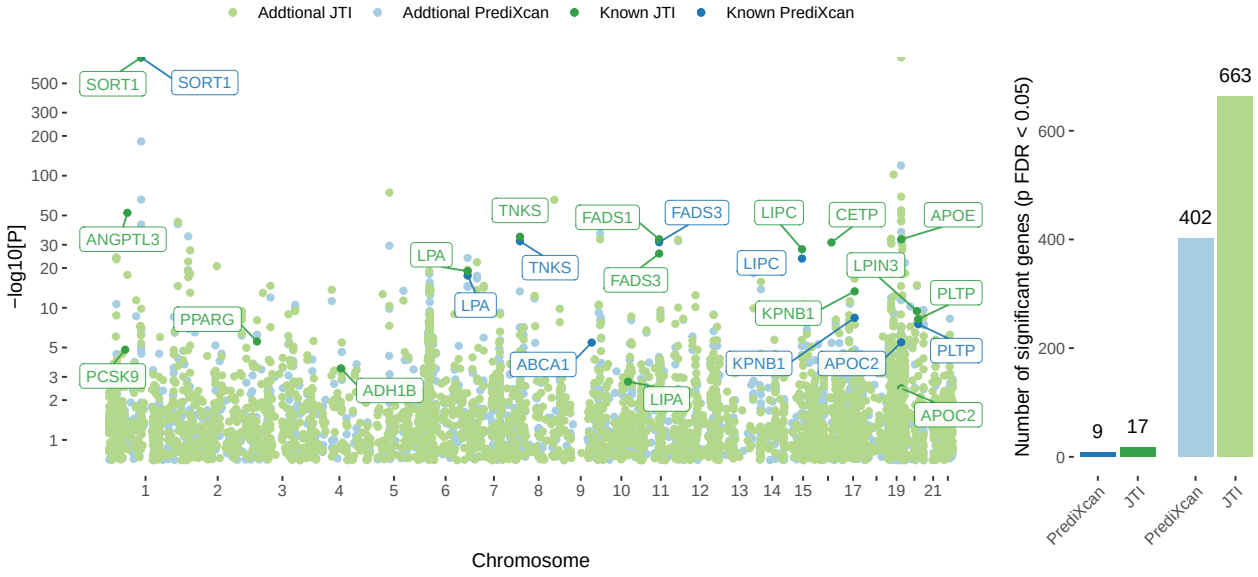
```
# Load known reference genes
known_genes <- read_sheet("S4_LDL_known_genes") %>% pull(genename)

# Load results from Zhou et. al Nat Gen 2020
zhou <- zhou_results() %>% mutate(known = genename %in% known_genes)

# Create lookup table for genes
genes <- zhou %>% distinct(geneid, genename, known, chr, bp)

# Load association results for LDL & Liver
ldl <- read_results(ukbb_id = "30780_irnt", tissue = "Liver") %>% inner_join(genes)
```

```
fig4(ldl %>% filter(method != "UTMOST"), title = "Manz 2021") / fig4(zhou, title = "Zhou et. al, Nature
```

Manz 2021



Zhou et. al, Nature Gen 2020

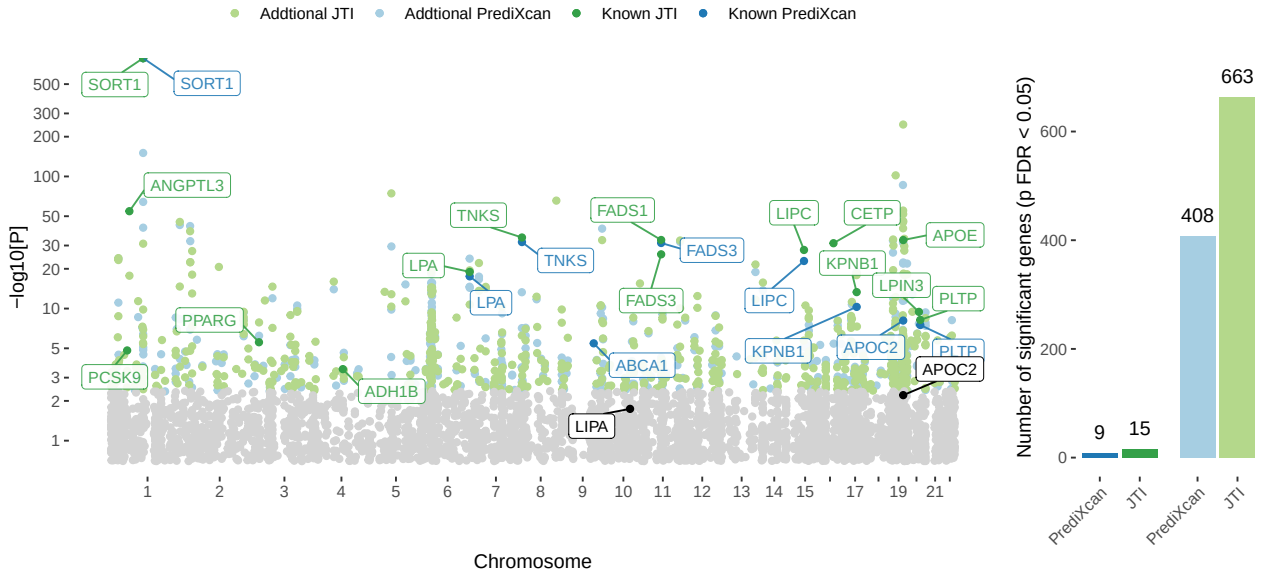


1.2 Annotate known genes that don't match Zhou et al.

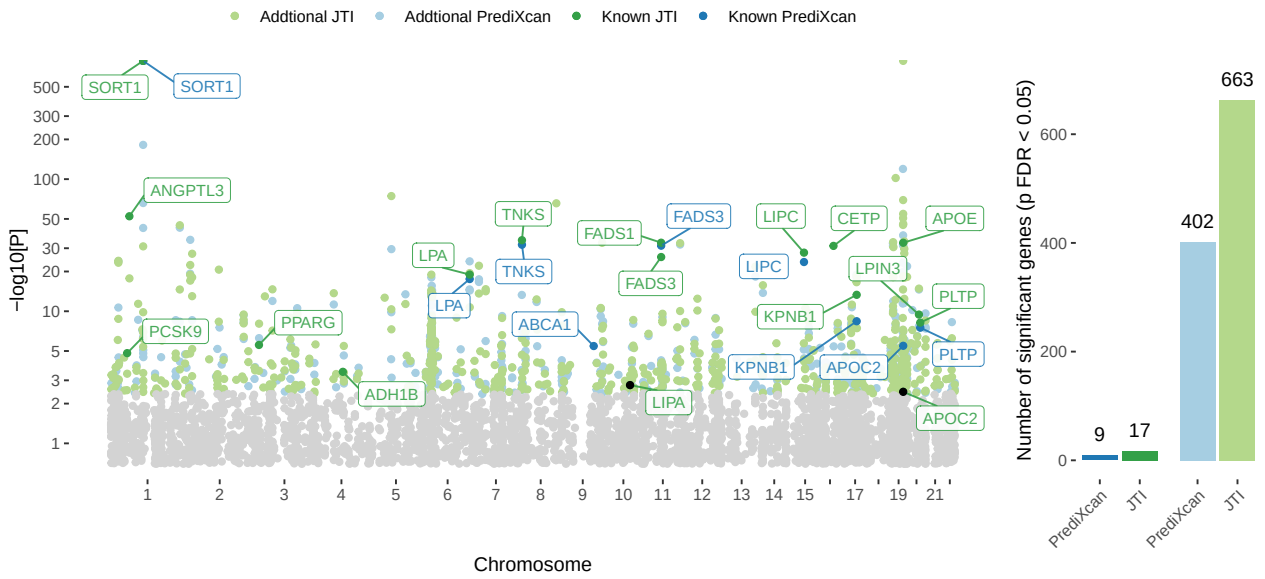
```
filter_missing <- function(x) filter(x, method == "JTI" & genename %in% c("LIPA", "APOC2"))

manz2 <- (manhatten_labeled_fdr(ldl %>% filter(method != "UTMOST"), gray_below = TRUE) +
  geom_label_repel(data = filter_missing, aes(label = genename), size = 3, show.legend = FALSE, box.pad
  geom_point(data = filter_missing) +
  ggtitle("Manz 2021")
) + sig_genes_bars(ldl %>% filter(method != "UTMOST")) + plot_layout(widths = c(5, 1))

zhou2 <- (manhatten_labeled_fdr(zhou, gray_below = TRUE) +
  geom_point(data = filter_missing) +
  ggtitle("Zhou et. al, Nature Gen 2020")
```

```
) + sig_genes_bars(zhou) +
  plot_layout(widths = c(5, 1))

manz2 / zhou2
```

Manz 2021

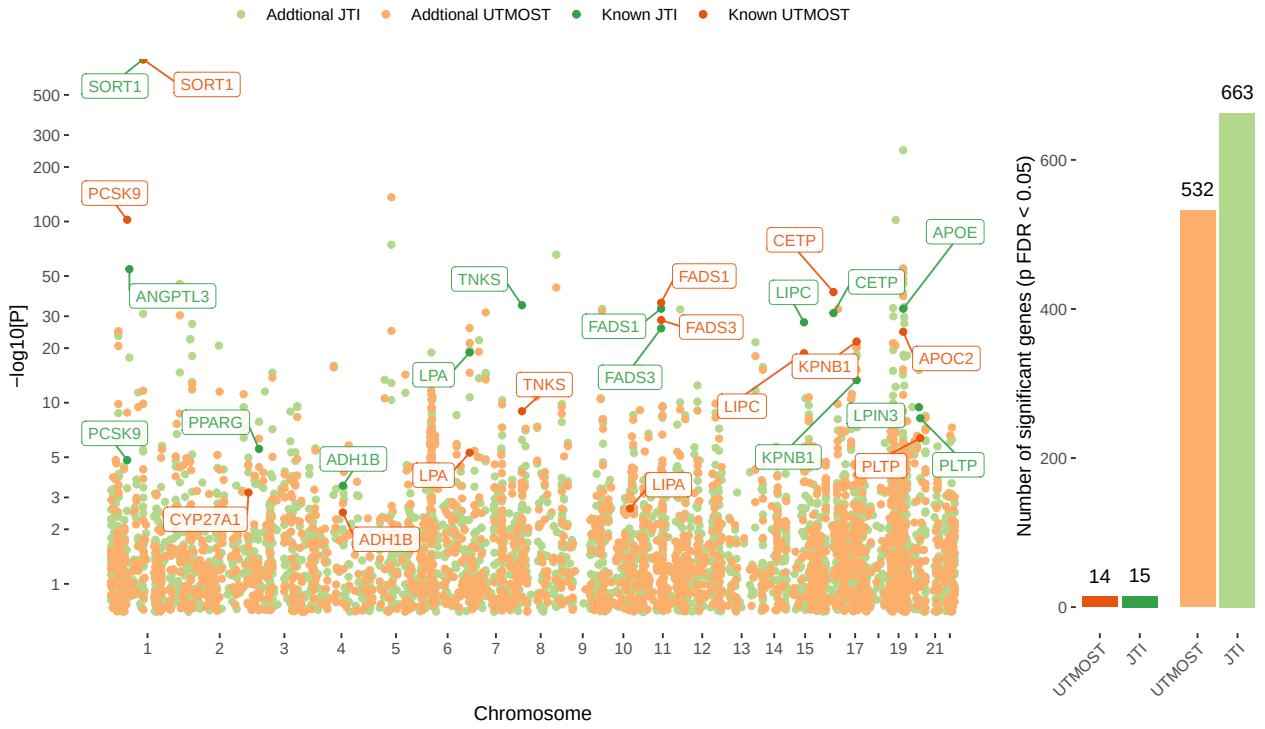


Zhou et. al, Nature Gen 2020

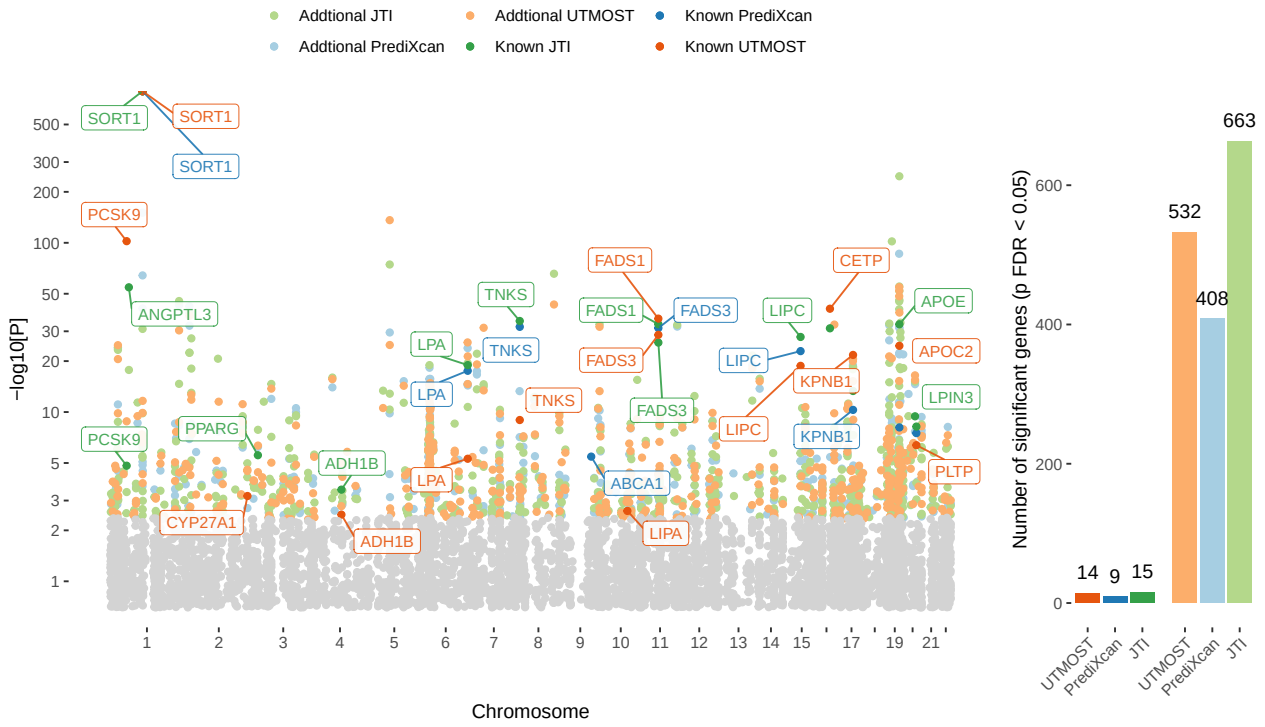


1.3 Compare with JTI with UTMOST

```
fig4(ldl %>% filter(method != "PrediXcan"), colors = c("#B4D88B", "#FCAE6B", "#34A048", "#E6550E"))
```




```
fig4(
  results = ldl,
  colors = c("#B4D88B", "#A6CEE2", "#FCAE6B", "#34A048", "#1F78B4", "#E6550E"),
  gray_below = TRUE
)
```



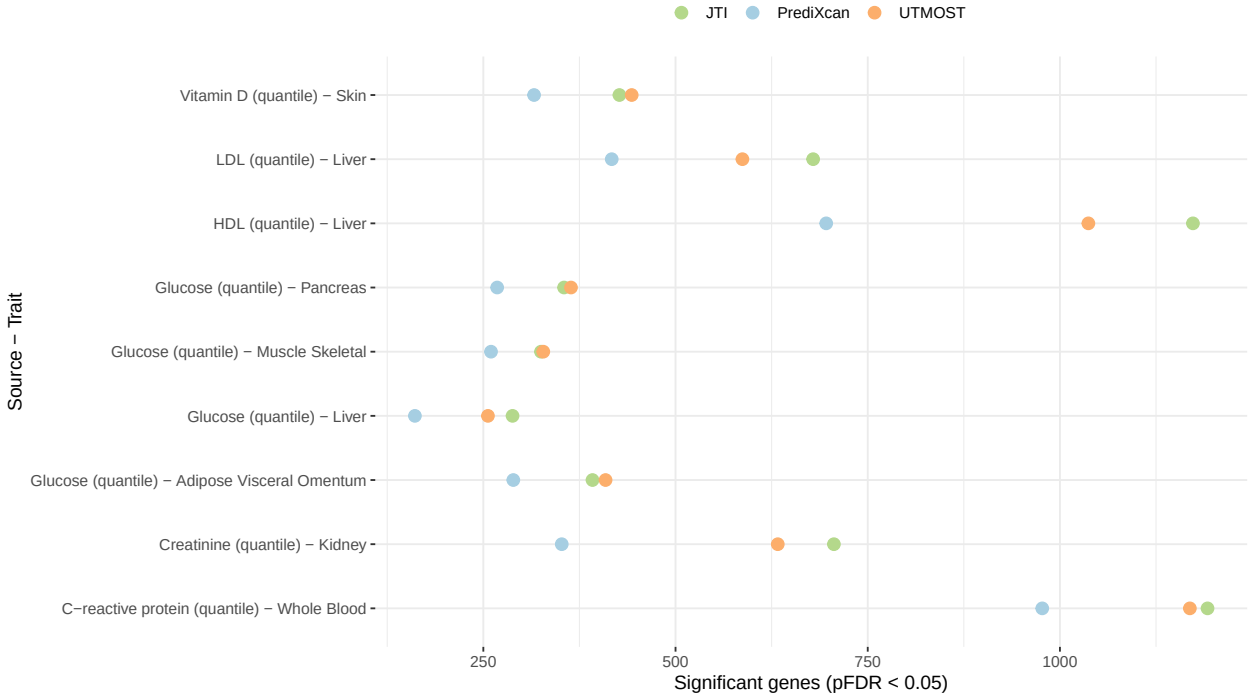
2 Comparison of significant genes in JTI, UTMOST, and PrediXcan

2.1 JTI, PrediXcan, and UTMOST performance comparison on additional GWAS datasets

```
library(tidyverse)

# Load local GWAS result
read_result <- function(ukbb_id, method, tissue) {
  fs::path("..", "results", paste0(ukbb_id, "-", method, "_", tissue, ".csv")) %>%
    read_csv() %>%
    # Match columns to zhou results
    rename(geneid = gene, genename = gene_name) %>%
    # Add new cols
    mutate(method = method, pfdr = p.adjust(pvalue, method = "fdr"))
}

# Merge results from multiple methods
read_results <- function(ukbb_id, tissue, methods = c("PrediXcan", "JTI", "UTMOST")) {
  methods %>%
    map(function(m) read_result(ukbb_id, m, tissue)) %>%
    bind_rows()
}

count_sig_genes <- function(ukbb_id, tissue, name) {
  read_results(ukbb_id, tissue) %>%
    filter(pfdr < 0.05) %>%
    count(method) %>%
    mutate(name = name)
}

gwas_summary <- bind_rows(
  count_sig_genes("30740_irnt", "Adipose_Visceral_Omentum", "Glucose (quantile) - Adipose Visceral Omen
  count_sig_genes("30740_irnt", "Liver", "Glucose (quantile) - Liver"),
  count_sig_genes("30740_irnt", "Muscle_Skeletal", "Glucose (quantile) - Muscle Skeletal"),
  count_sig_genes("30740_irnt", "Pancreas", "Glucose (quantile) - Pancreas"),
  count_sig_genes("30760_irnt", "Liver", "HDL (quantile) - Liver"),
  count_sig_genes("30780_irnt", "Liver", "LDL (quantile) - Liver"),
  count_sig_genes("30890_irnt", "Skin_Sun_Exposed_Lower_leg", "Vitamin D (quantile) - Skin"),
  count_sig_genes("30710_irnt", "Whole_Blood", "C-reactive protein (quantile) - Whole Blood"),
  count_sig_genes("30700_irnt", "Kidney_Cortex", "Creatinine (quantile) - Kidney")
)

ggplot(gwas_summary, aes(n, name)) +
  geom_point(aes(color = method), size = 3) +
  theme_minimal() +
  scale_color_manual(values = c("#B4D88B", "#A6CEE2", "#FCAE6B")) +
  theme_bw() +
  theme(
    panel.border = element_blank(),
    legend.position = "top",
    plot.title.position = "plot"
```

```
  ) +
  labs(x = "Significant genes (pFDR < 0.05)", y = "Source - Trait", color = "") +
  ggtitle("JTI, PrediXcan, and UTMOST performance comparison on additional GWAS datasets")
```

JTI, PrediXcan, and UTMOST performance comparison on additional GWAS datasets



3 Comparison of weights in JTI, UTMOST, and PrediXCan

3.1 Summary of weight vector statistics by gene and weights across all genes and methods

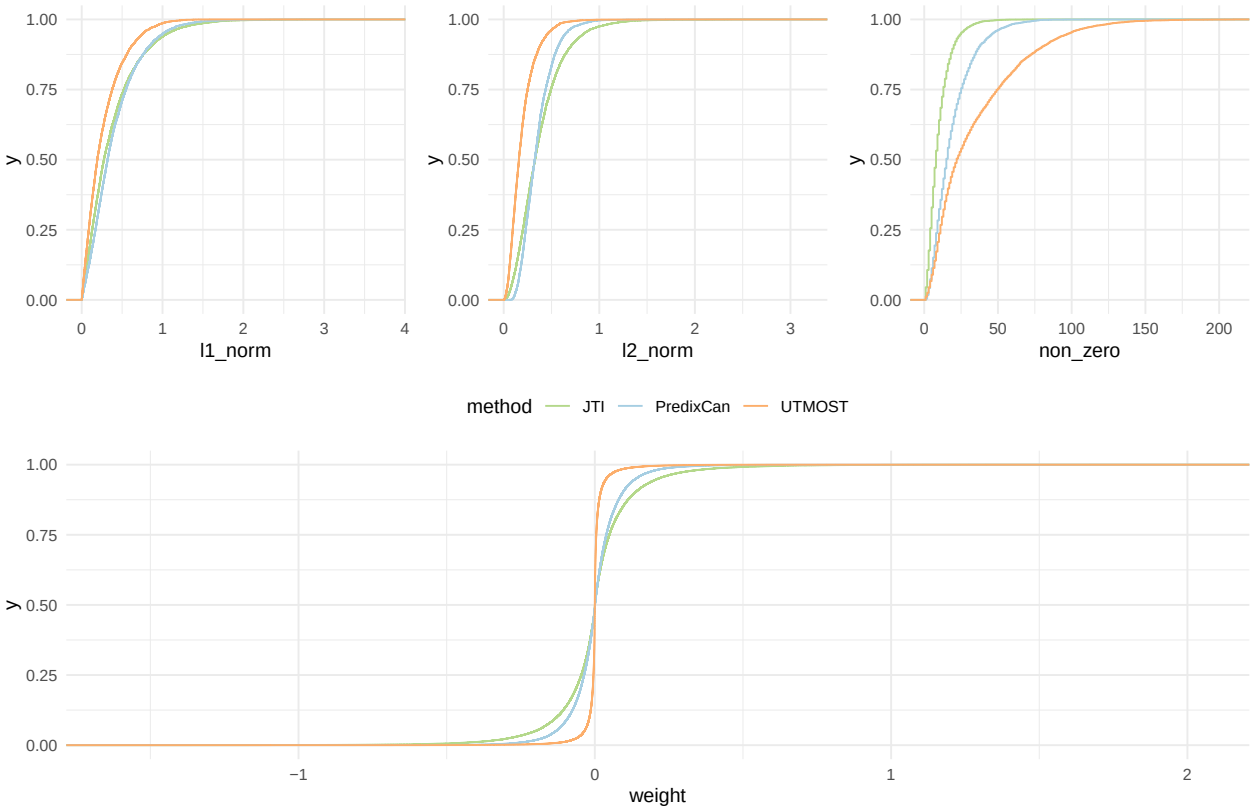
```
library(tidyverse)
library(patchwork)
library(GGally)

load_weights <- function(method, tissue) {
  db <- fs::path("..", "data", "weights", paste0(method, "_", tissue, ".db"))
  conn <- DBI::dbConnect(RSQLite::SQLite(), db)
  df <- tbl(conn, "weights") %>% collect()
  DBI::dbDisconnect(conn)
  df %>% mutate(method = method)
}

load_tissue <- function(tissue, methods = c("UTMOST", "PredixCan", "JTI")) {
  methods %>%
    map(function(m) load_weights(m, tissue)) %>%
    bind_rows()
}

summarize_weights <- function(tissue_df) {
```

```
  tissue_df %>%
    group_by(gene, method) %>%
    summarise(
      l1_norm = abs(sum(weight)),
      l2_norm = sqrt(sum(weight^2)),
      non_zero = n(),
      .groups = "drop"
    )
}

plot_tissue_summary <- function(tissue) {
  title <- paste("Distribution of L1 norm, L2 norm, and sparcity of gene weights for", tissue)
  weights <- load_tissue(tissue)
  summary <- weights %>% summarize_weights()

  color_s <- scale_color_manual(values = c("#B4D88B", "#A6CEE2", "#FCAE6B"))
  base <- ggplot(summary, aes(color = method)) +
    color_s +
    guides(color = FALSE) +
    theme_minimal()

  p1 <- base + stat_ecdf(aes(l1_norm))
  p2 <- base + stat_ecdf(aes(l2_norm))
  p3 <- base + stat_ecdf(aes(non_zero))
  p4 <- ggplot(weights, aes(weight, color = method)) +
    stat_ecdf(aes(weight)) +
    color_s +
    theme_minimal() +
    theme(legend.position = "top")

  plt <- (p1 | p2 | p3) / p4
  plt + plot_annotation(
    title = tissue,
    subtitle = "ECDFs of weight vector statistics by gene (top) and all weights (bottom)"
  )
}
```

```
plot_tissue_summary("Liver")
```

Liver

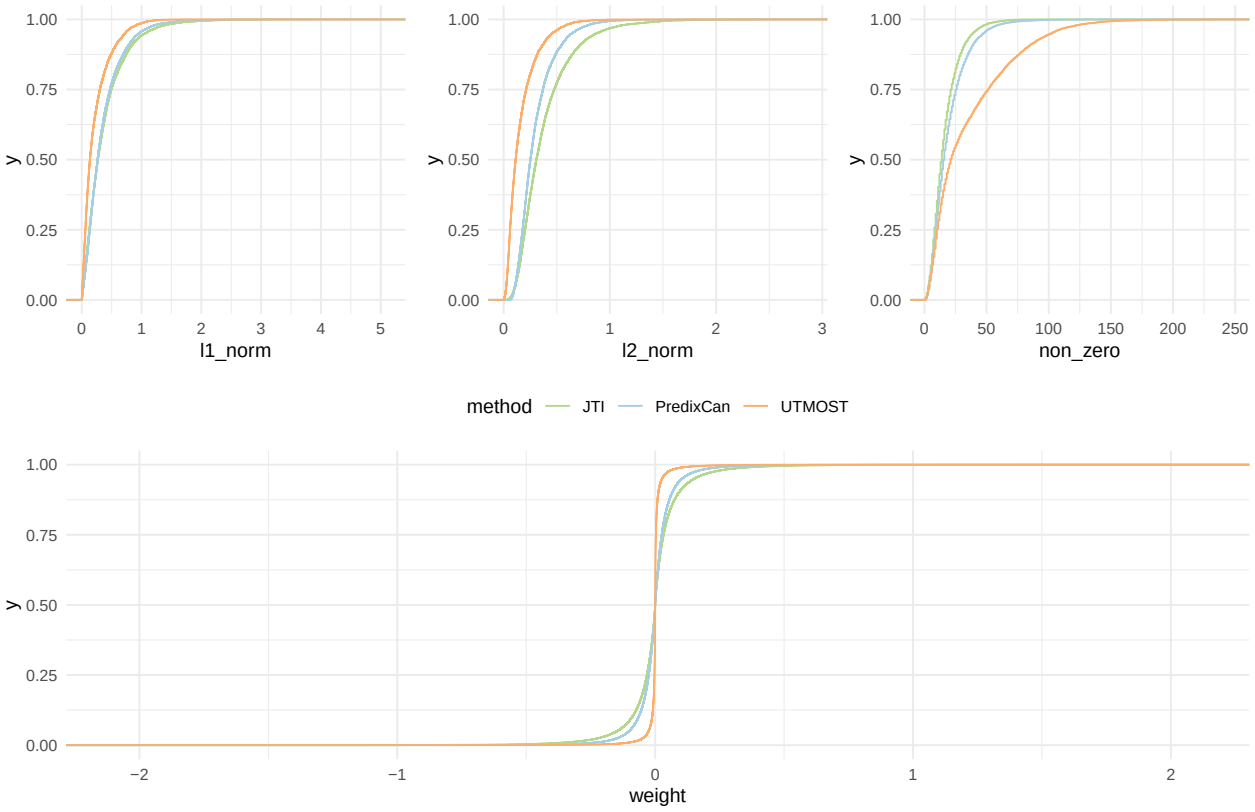
ECDFs of weight vector statistics by gene (top) and all weights (bottom)




```
plot_tissue_summary("Muscle_Skeletal")
```

Muscle_Skeletal

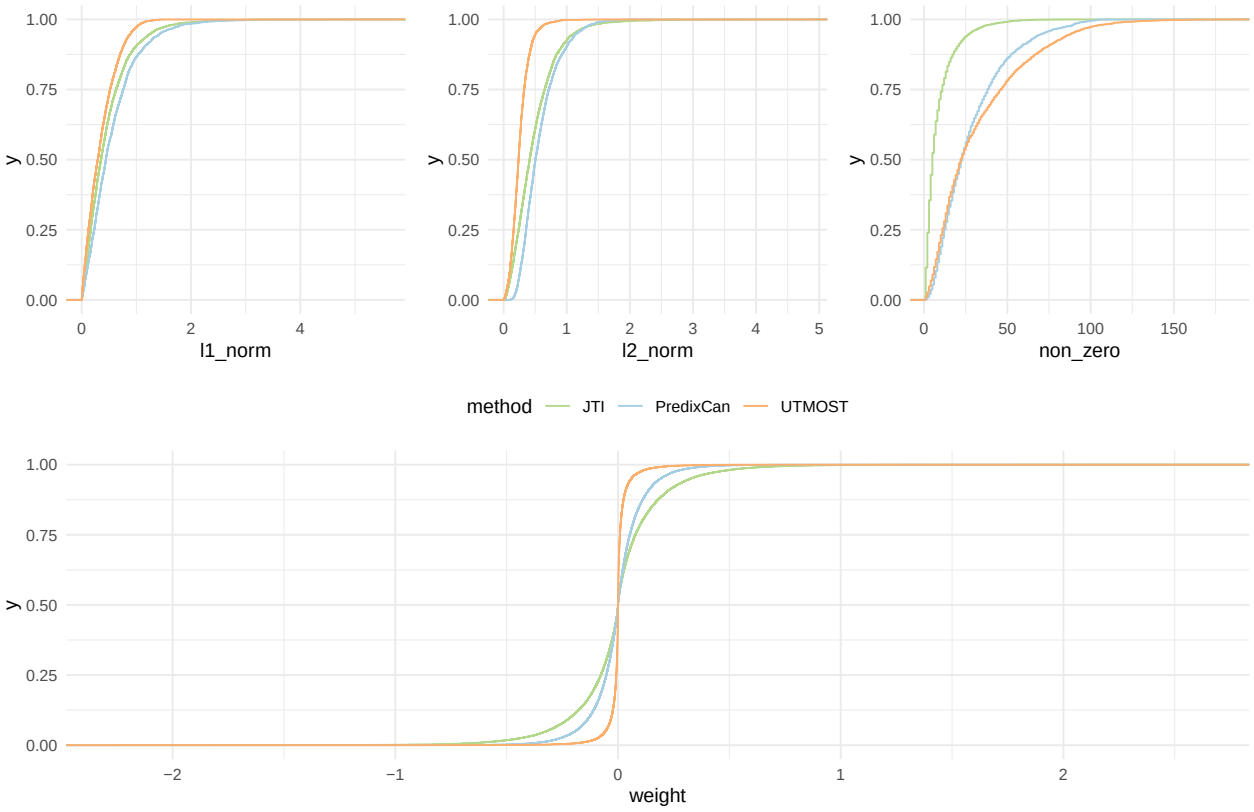
ECDFs of weight vector statistics by gene (top) and all weights (bottom)




```
plot_tissue_summary("Kidney_Cortex")
```

Kidney_Cortex

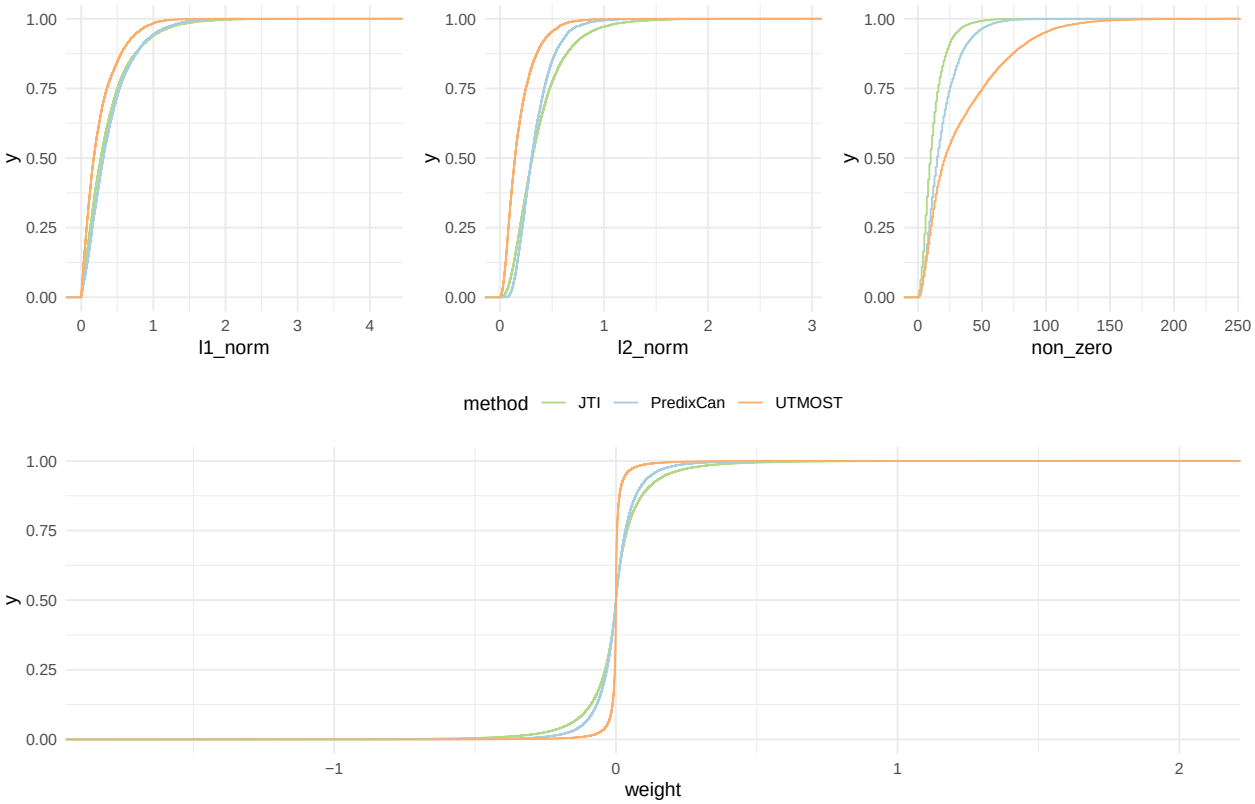
ECDFs of weight vector statistics by gene (top) and all weights (bottom)




```
plot_tissue_summary("Pancreas")
```

Pancreas

ECDFs of weight vector statistics by gene (top) and all weights (bottom)



3.2 Element-wise comparison of weight vectors

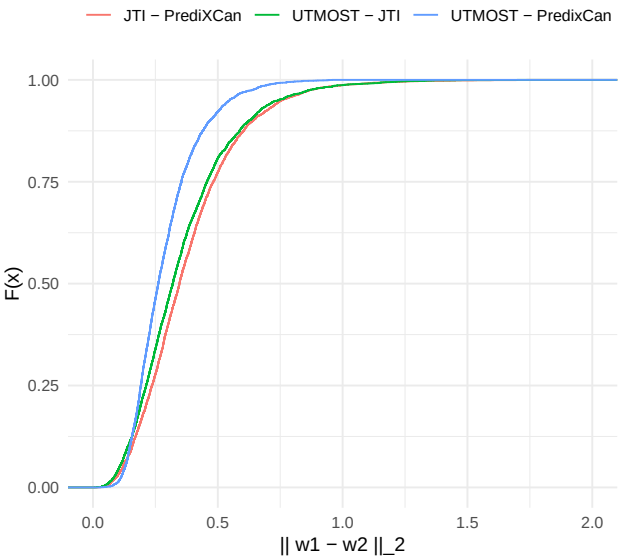
```r
load_summary <- function(tissue) {
  # Load all weights from DBs
  weights <- load_tissue(tissue) %>%
    select(rsid, gene, weight, method)

  # Find genes for which all three methods have weights
  genes <- weights %>%
    select(gene, weight, method) %>%
    group_by(method, gene) %>%
    pivot_wider(
      names_from = method,
      values_from = weight,
      values_fn = length
    ) %>%
    drop_na() %>%
    pull(gene)

  # Compute the l2 norm of diff between weights for different methods
  l2_diff <- weights %>%
    filter(gene %in% genes) %>%
    pivot_wider(names_from = method, values_from = weight, values_fill = 0) %>%
    group_by(gene) %>%
    summarise(
```

```
      UTMOST_PredixCan = norm(UTMOST - PredixCan, type = "2"),
      UTMOST_JTI = norm(UTMOST - JTI, type = "2"),
      JTI_PrediXCan = norm(JTI - PredixCan, type = "2")
    ) %>%
    pivot_longer(
      -gene,
      names_to = c("method1", "method2"),
      names_sep = "_",
      values_to = "diff_l2"
    ) %>%
    mutate(tissue = tissue)
}

plot_diff_summary <- function(tissue_diff_summary, title) {
  tissue_diff_summary %>%
    ggplot(aes(diff_l2, color = paste(method1, method2, sep = " - "))) +
    stat_ecdf() +
    theme_minimal() +
    theme(
      legend.position = "top",
      legend.title = element_blank(),
      plot.title.position = "plot"
    ) +
    labs(x = "|| w1 - w2 ||_2", y = "F(x)") +
    scale_x_continuous(limits = c(0, 2))
}

plot_comparison <- function(t1, t2, t3, t4) {
  p1 <- load_summary(t1) %>% plot_diff_summary() + ggtitle(t1)
  p2 <- load_summary(t2) %>% plot_diff_summary() + ggtitle(t2)
  p3 <- load_summary(t1) %>% plot_diff_summary() + ggtitle(t3)
  p4 <- load_summary(t2) %>% plot_diff_summary() + ggtitle(t4)
  plt <- (p1 + p2) / (p3 + p4)
  plt + plot_annotation(
    title = "Element wise comparison of weight vectors across all genes"
  )
}
```

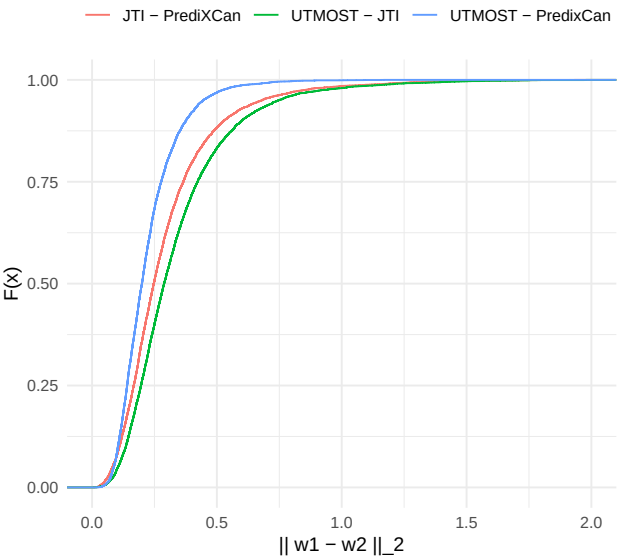
```
plot_comparison("Liver", "Muscle_Skeletal", "Kidney_Cortex", "Pancreas")
```

Element wise comparison of weight vectors across all genes

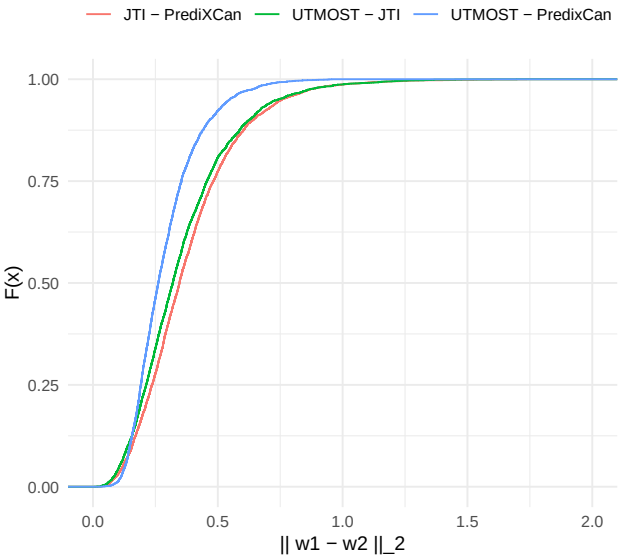
Liver



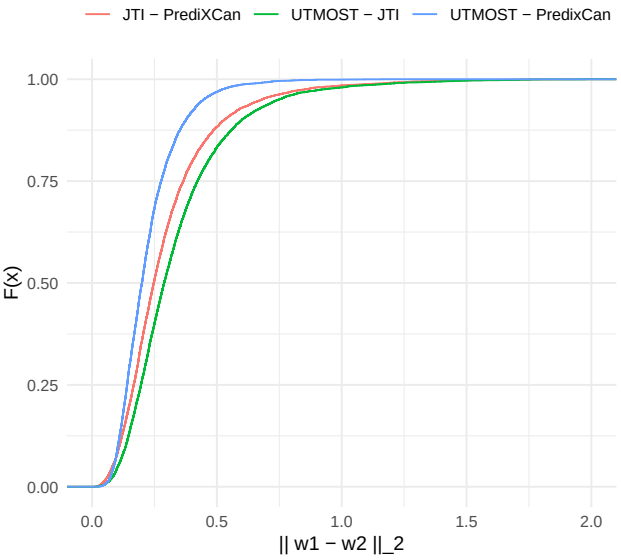
Muscle_Skeletal



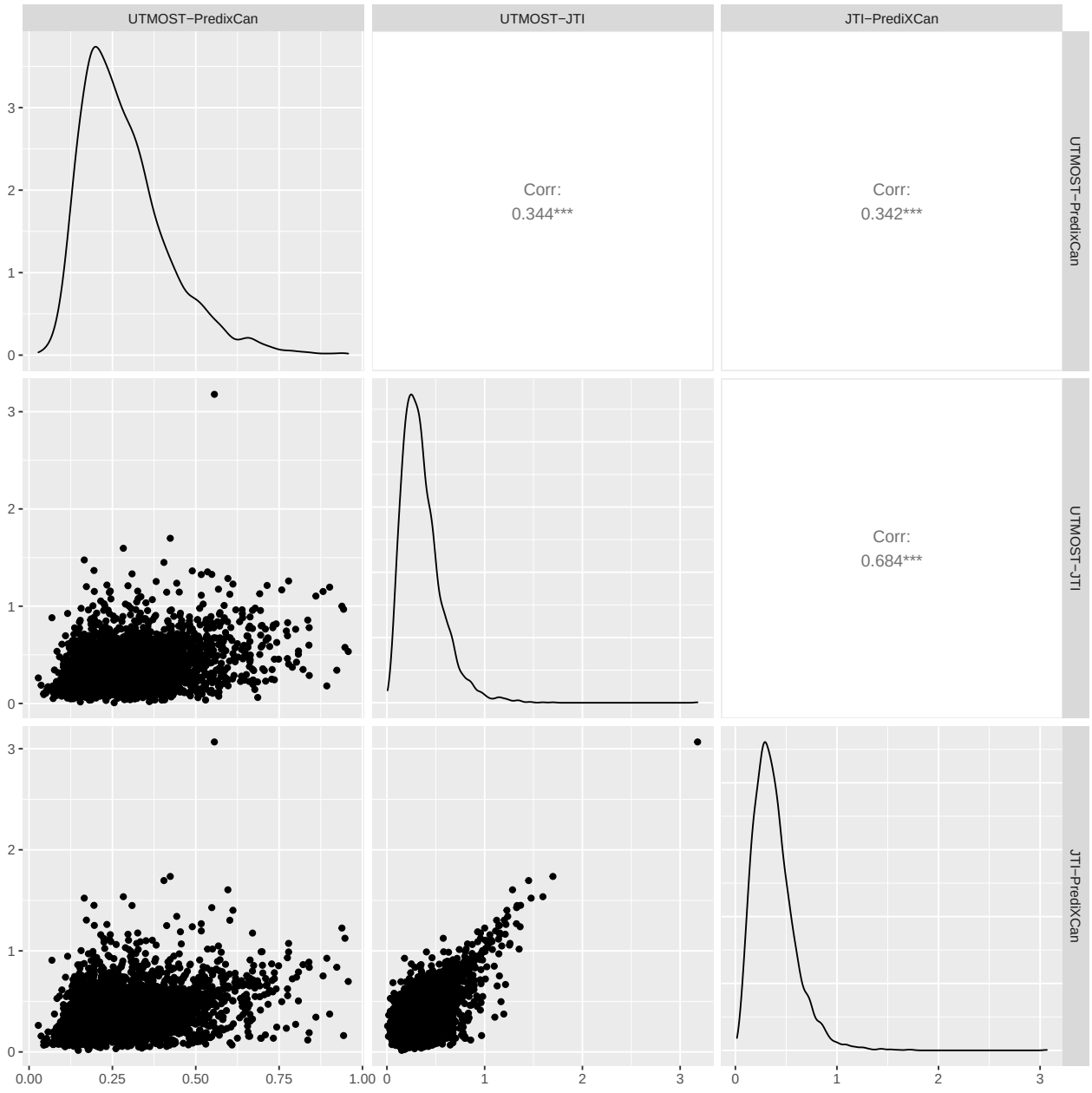
Kidney_Cortex



Pancreas




```
load_summary("Liver") %>%
  pivot_wider(
    names_from = c(method1, method2),
    names_sep = "-",
    values_from = diff_l2
  ) %>%
  drop_na() %>%
  select(-gene, -tissue) %>%
  ggpairs()
```




```
load_summary("Muscle_Skeletal") %>%
  pivot_wider(
    names_from = c(method1, method2),
    names_sep = "-",
    values_from = diff_l2
  ) %>%
  drop_na() %>%
  select(-gene, -tissue) %>%
  ggpairs() +
  ggtitle("Muscle Skeletal")
```

Muscle Skeletal