Olney Corridor Analysis

Zoe Yoo

12/6/2021

Contents

Overall Corridor Analysis

Broad Descriptive Stats

The Olney transit corridor runs along Olney Ave between Chelten Avenue and Front Street. The 18 and 26 are the two routes that run along the corridor, both crossing North Broad Street past the Olney Transportation Center. The Transportation Center has a stop for the Broad Street Line, while there is also a stop for the Fox Station Regional Rail line at the end of the corridor adjacent to Olney & Front Street. There are 49 total bus stops along the corridor, with the average bus trip running past 22 stops spanning 2.3 miles.

- ~~Factors we have: average segment speed, stop spacing (avg_stop_spacing_ft), ons/offs, average load, max entry load (max # of people on first? load), dwell time~~
- ~~Map of corridor? Automated using leaflet? maybe kind of tricky~~

```r
stop_list <- I(list(c("SEPTA372","SEPTA15911","SEPTA15794", "SEPTA15912", "SEPTA15789", "SEPTA15915", "S
subcorridor_name <- c("Olney and Chelten - Olney and Broad", "Olney and Broad - Olney and 7th", "Olney a
subcorridor_id <- c("CB","B7","7F")
subcorridors <- data.frame(subcorridor_id,subcorridor_name,stop_list)

stop_list <- unlist(stop_list)

#create_stops()
stops_w_ridership <- st_read("../data/stops_ridership.geojson", quiet = TRUE) %>%
    mutate_if(is.factor, as.character) %>%
    mutate(route_list = as.list(strsplit(as.character(route_str), ",")))
#create_routes()
routes_w_ridership <- st_read("../data/routes_ridership.geojson", quiet = TRUE) %>%
    mutate_if(is.factor, as.character)
# import links
links <- st_read("../data/links_with_stops.geojson", quiet = TRUE)
link_stop_data <- links %>% mutate(fromto = as.character(fromto), secondLocationID = as.character(second
    mutate(stop_list = as.list(strsplit(as.character(stop_str), ",")))
# import APC data
apc_data <- read.csv("../data/combined_apc_dataset.csv") %>%
```

```
    mutate(stop_id = paste0(agency_id, stop_id)) %>%
    mutate(dwell_time = case_when(
        agency_id == "NJT" ~ dwell_time * 60,
        TRUE ~ dwell_time
    ))
apc_data <- adjust_dwell_and_velo(apc_data)

alltracts <- st_read("https://opendata.arcgis.com/datasets/8bc0786524a4486bb3cf0f9862ad0fbf_0.geojson",
#restrict to just tracts that stops are in


#pulling the acs data
acs_trans <- c("B08301_001", #workers
               "B08301_002", #means of transportation to work: car
               "B08301_003", #means of transportation to work: car, drove alone
               "B08301_010", #means of transportation to work: public transit
               "B08301_011", #means of transportation to work: public transit, bus
               "B08301_012", #means of transportation to work: public transit, subway
               "B08301_013", #means of transportation to work: public transit, commuter rail
               "B08301_014", #means of transportation to work: public transit, light rail/trolley
               "B08301_016", #means of transportation to work: taxicab
               "B08301_018", #means of transportation to work: bicycle
               "B08301_019", #means of transportation to work: walked
               "B08301_021" #means of transportation to work: wfh
               )

trip_dat <- (find_trip_dat_v2(apc_data, stop_list))


## [1] "Running passenger data for 464 trips"

analytics <- analyze_segment(trip_dat)
route_analytics <- analyze_segment_route(trip_dat)
binned_analytics <- suppressWarnings(analyze_segment_hourbin(trip_dat))
hourly_analytics <- suppressWarnings(analyze_segment_hourly(trip_dat))
hourly_route_analytics <- suppressWarnings(analyze_segment_route_hourly(trip_dat))
hourly_route_direction_analytics <- suppressWarnings(analyze_segment_route_direction_hourly(trip_dat))

routes <- subset(routes_w_ridership, routes_w_ridership$Route %in% trip_dat$route_id) %>%


# map of routes that exist in this corridor
leaflet() %>%
  setView(lng = -75.14511002073658, lat = 40.03905345107288, zoom = 12) %>%
  addProviderTiles(providers$Stamen.Toner) #%>%
  #addPolylines(data = routes, color = "#4377bc", weight = 4, layerId = link_stop_data$fromto, opacity
```

**Corridor Level Descriptive Stats**

These charts illustrate characteristics for the Olney corridor as a whole, such as average speed and ridership.
Average speed, as well as riderhship, vary both by route and by time of day.

route_id	daily_ridership	trips	routes_served	service_hours	riders_per_hour	on_off	dwell_observed_mean	dwell_predicted_mean	dwell_hybrid_mean	dwell_per_onoff	onoff_per_trip	onoff_per_tripstop	avg_segment_speed	avg_speed_10_pct	avg_speed_25_pct	avg_speed_75_pct	avg_speed_90_pct
18	10,592	235	18	30.1	351.6	11,448.4	0	216.0	216.0	0.2	48.7	2.1	11.2	9.1	9.6	11.9	14.2
26	8,414	229	26	55.5	151.5	9,839.4	0	199.3	199.3	0.2	43.0	1.8	10.9	9.4	9.9	11.6	13.2
Total	19,005	464	18, 26	85.7	221.9	21,287.8	0	207.8	207.8	0.2	45.9	1.9	11.0	9.2	9.8	11.7	14.0

```r
table_1 <- route_analytics %>% bind_rows(analytics %>% mutate(route_id = "Total"))

kable(table_1, booktabs = TRUE, align = 'c',format.args = list(big.mark = ","),digits=1) %>%
  kable_styling(latex_options = "scale_down")  %>%
  row_spec(dim(table_1)[1], bold = T) %>% # format last row
  column_spec(1, italic = T) %>%  # format first column
  scroll_box(width = "100%", height = "300px")
```
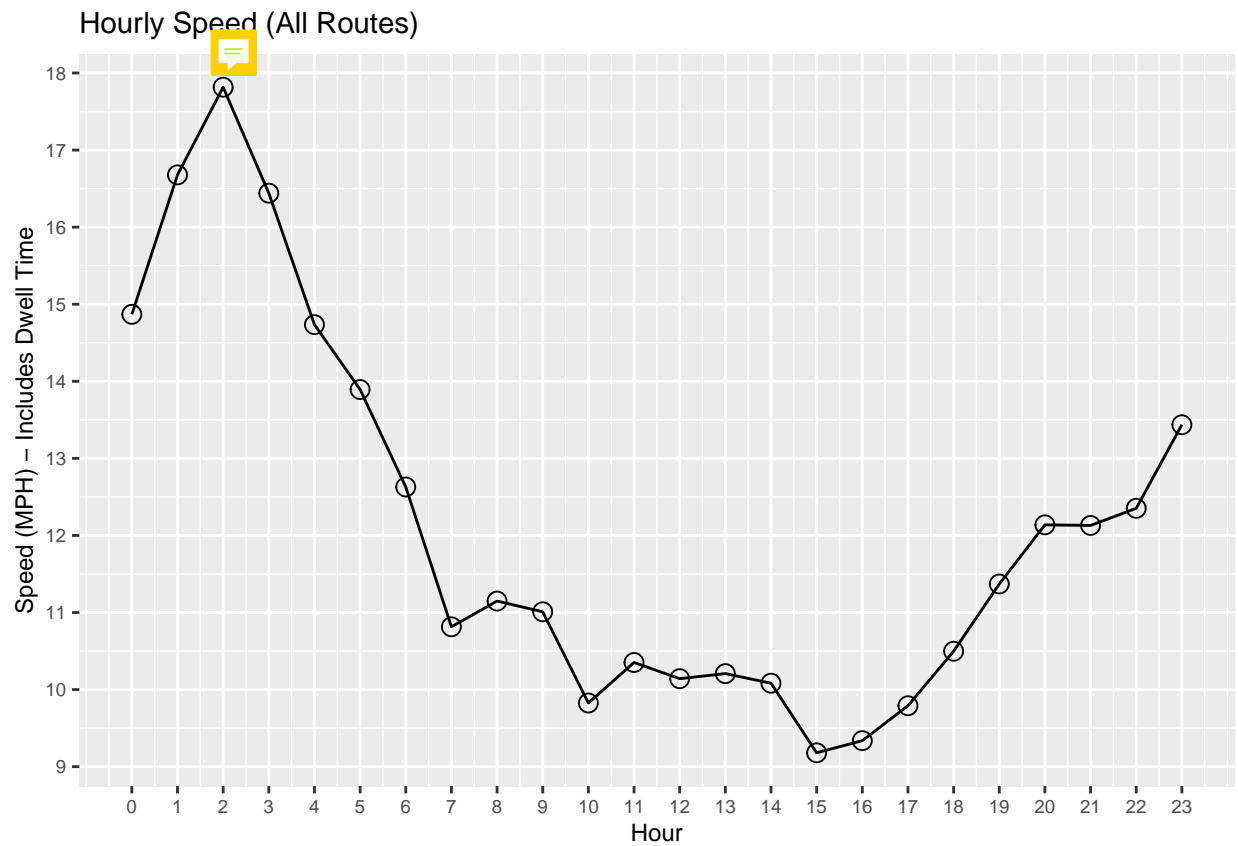
```r
ggplot(route_analytics, aes(x = route_id, y = avg_segment_speed)) +
  geom_bar(stat = "identity", fill="skyblue", alpha=0.7) +
  geom_errorbar(data = route_analytics, stat = "identity", ymin = route_analytics$avg_speed_10_pct, ymax
  scale_y_continuous(name = "Speed (MPH) - Includes Dwell Time", n.breaks = 8, limits = c(0, max(route_a
  scale_x_discrete(name = "Route Number") +
  labs(title = paste0("The Average Bus Travels at ", round(mean(analytics$avg_segment_speed), 1), " MPH
  theme(text = element_text(size = 9))
```


The Average Bus Travels at 11 MPH on the Corridor
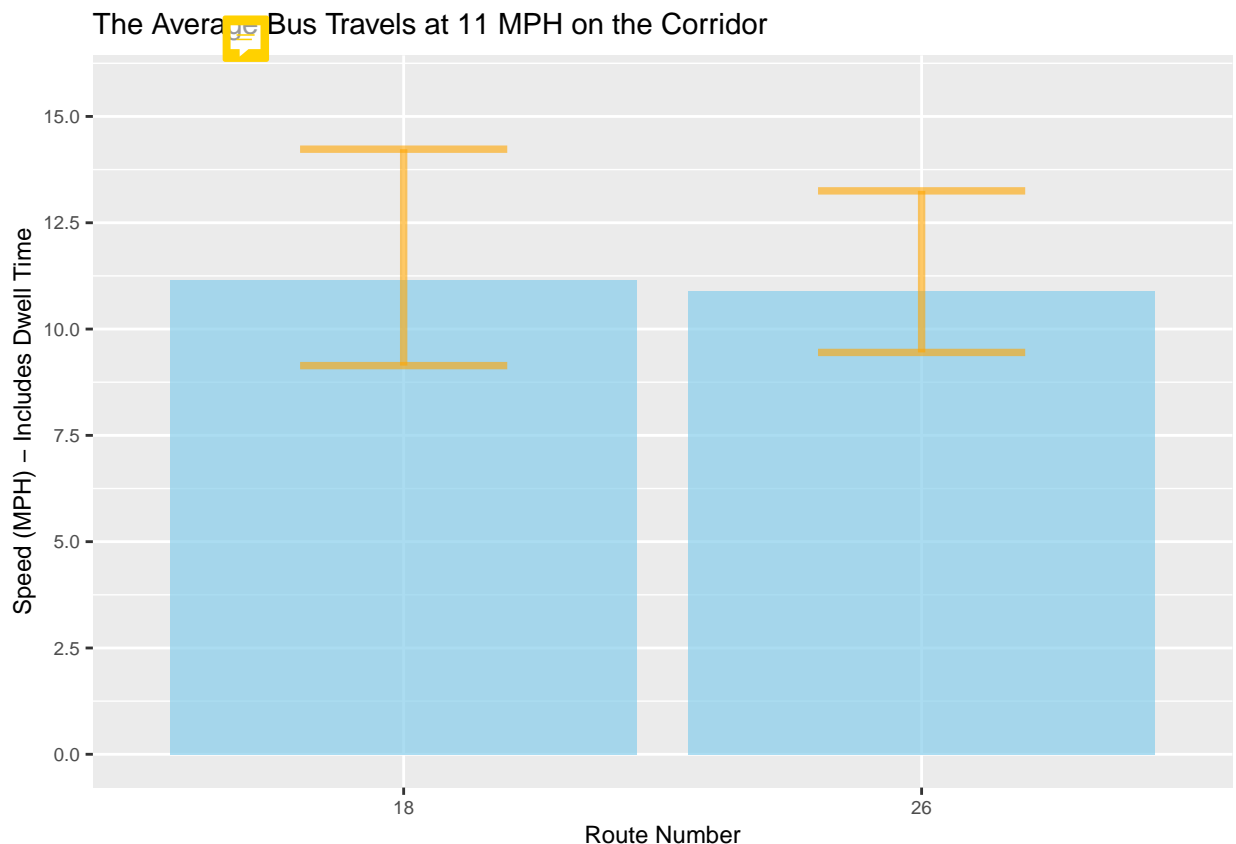
```r
ggplot(hourly_analytics) +
  geom_line(aes(x = trip_hour, y = avg_segment_speed)) +
  geom_point(aes(x = trip_hour, y = avg_segment_speed), shape=21, size=3) +
  scale_y_continuous(name = "Speed (MPH) - Includes Dwell Time", n.breaks = 8) +
  scale_x_continuous(name = "Hour", breaks = c(0:23)) +
```

```
labs(title = "Hourly Speed (All Routes)")+
theme(text = element_text(size = 9))
```

Hourly Speed (All Routes)


```
ggplot(binned_analytics, aes(x = timeframe, y = avg_segment_speed)) +
  geom_bar(stat = "identity", fill="skyblue", alpha=0.7) +
  geom_errorbar(data = binned_analytics, stat = "identity", ymin = binned_analytics$avg_speed_10_pct, ym
  scale_y_continuous(name = "Speed (MPH) - Includes Dwell Time", n.breaks = 8, limits = c(0, max(binned_
  scale_x_discrete(name = element_blank()) +
  labs(title = "Speed by Period (All Routes)")+
  theme(text = element_text(size = 9))
```

Speed by Period (All Routes)
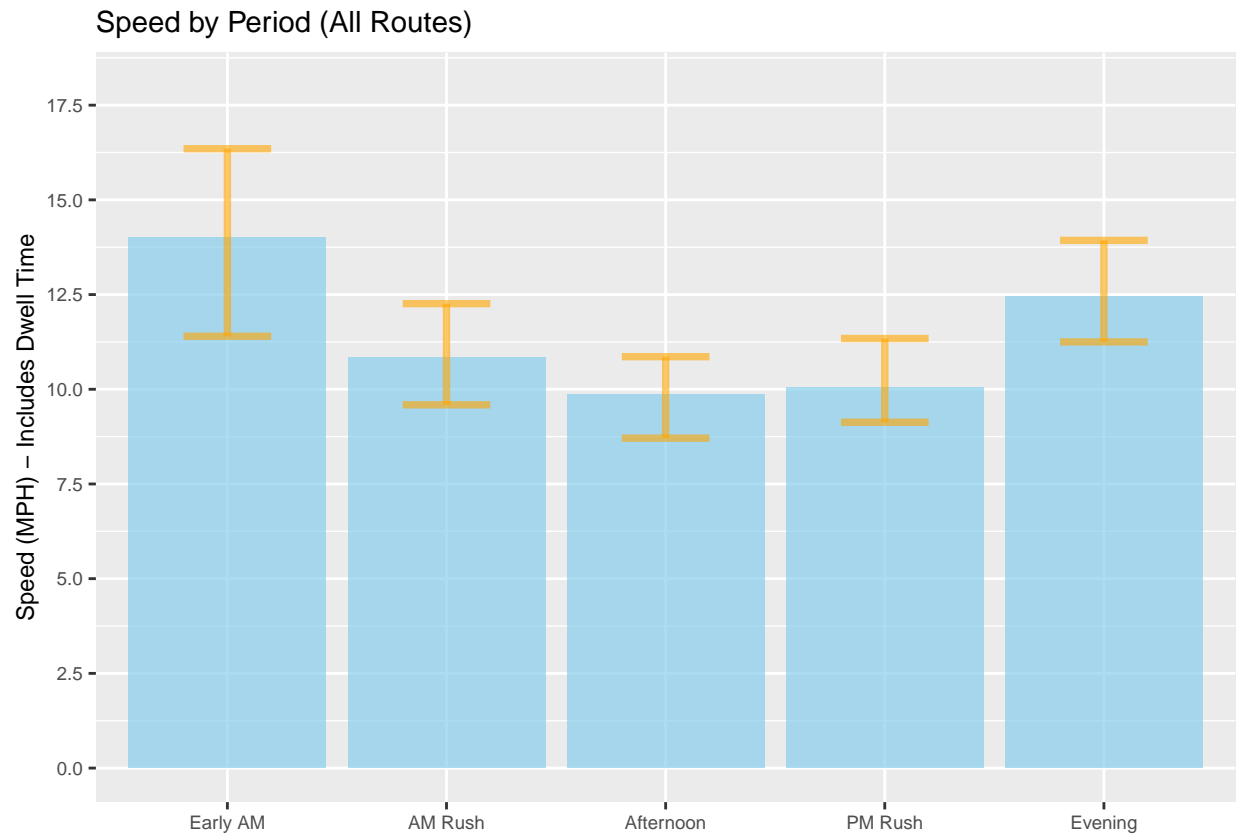

```
ggplot(binned_analytics, aes(x = timeframe, y = daily_ridership, label = daily_ridership)) +
  geom_bar(stat = "identity", fill="skyblue", alpha=0.7) +
  geom_text(position=position_dodge(width=0.9), vjust = -0.75, size = 3) +
  scale_y_continuous(name = "Total Riders Served", n.breaks = 5) + #, limits = c(0, max(binned_analytic
  scale_x_discrete(name = element_blank()) +
  labs(title = "Ridership by Period (All Routes)")+
  theme(text = element_text(size = 9))
```

Ridership by Period (All Routes)
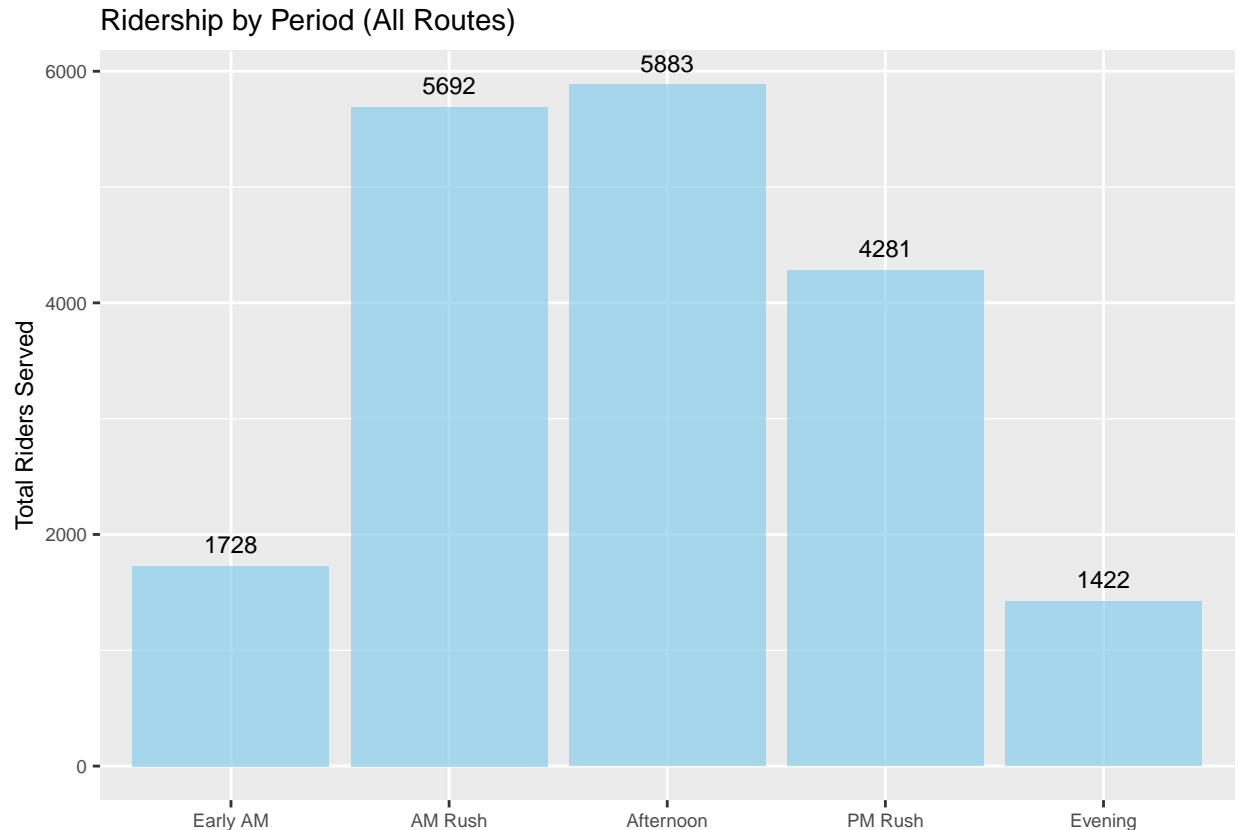

Corridor/Route Level Stats

Here, we separate overall corridor statistics by route to better understand the makeup of Olney bus traffic. The first three graphs, which cover ridership (both average daily ridership and average ridership per hour), the number of trips, and number of service hours per route, aggregate for both directions of each route. Next, average hourly ridership is again shown, but divided by whether the bus was Eastbound or Westbound (as Olney runs E-W). Average corridor running time, or the time it takes for a bus to make one trip along the corridor, and average hourly speed are also differentiated by direction.

```r
grid.arrange(ncol=2,
ggplot(route_analytics, aes(y = daily_ridership, x = route_id, label = daily_ridership)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  geom_text(position=position_dodge(width=0.9), vjust = -0.75, size = 3) +
  scale_y_continuous(name = "Total Riders Served", n.breaks = 8, limits = c(0, max(route_analytics$daily
  scale_x_discrete(name = "Route Number") +
  labs(title = paste0("The Corridor Serves ", round(sum(analytics$daily_ridership)), " Riders per Day"))
  theme(text = element_text(size = 9)),
ggplot(route_analytics, aes(x=route_id, y = trips, label = trips)) +
  geom_bar(position="dodge", stat="identity") +
  geom_text(position=position_dodge(width=0.9), vjust = -0.75, size = 3) +
  ylab("Daily Trips (2019)") +
  scale_x_discrete(name = "Route Number") +
  scale_fill_phl(palette = "main", discrete = T) +
  labs(fill= "", title = paste0("The Corridor Serves ", sum(route_analytics$trips, na.rm = TRUE), " Trip
  #theme_phl() +
```

6

```
    theme(legend.position = "top")+
    theme(text = element_text(size = 9))
)
```

The Corridor Serves 19005 Riders per [**The Corridor Serves 464 Trips per Day**
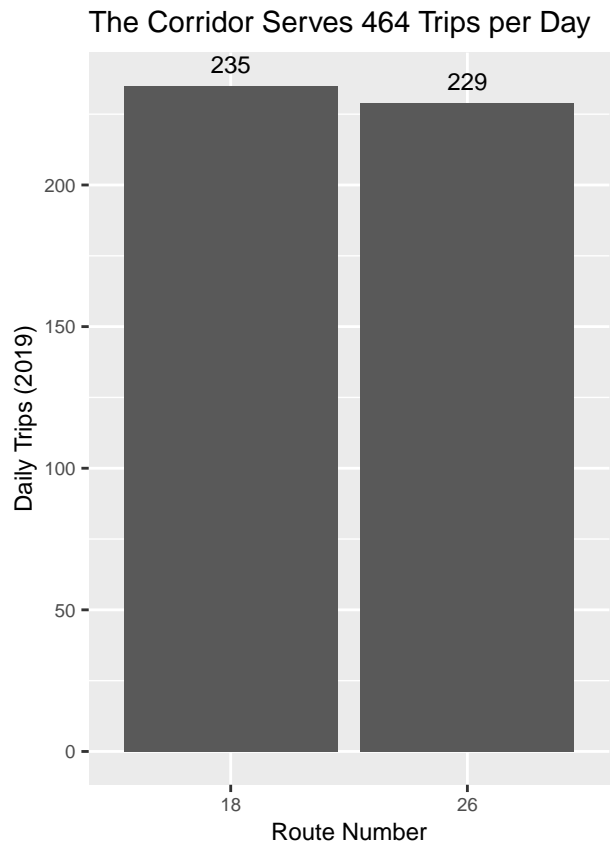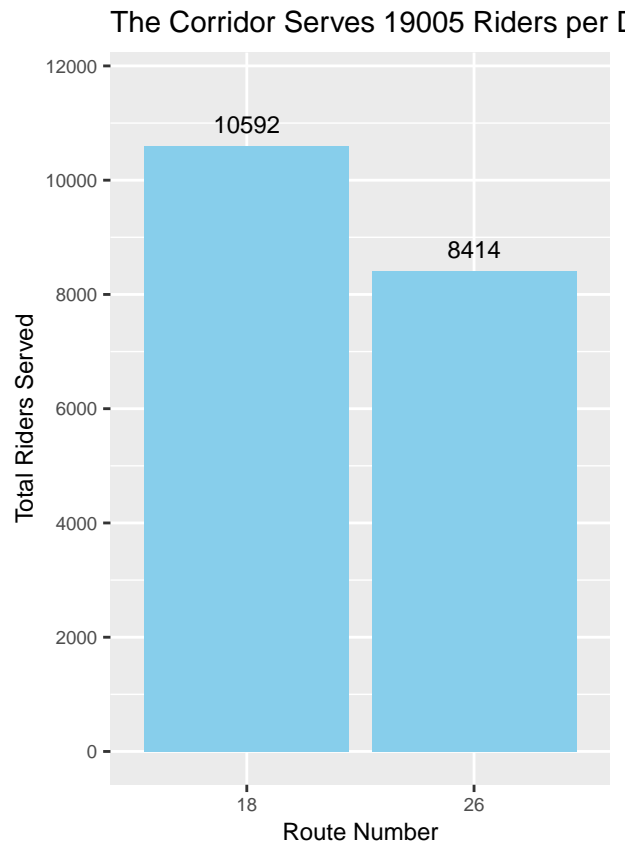


```
ggplot(route_analytics, aes(x=route_id, y = service_hours, label = round(service_hours))) +
  geom_bar(position="dodge", stat="identity") +
  geom_text(position = "dodge", stat = "identity", vjust = -0.25) +
  #ggtitle("Market and JFK serve over 1,100 Bus Trips per Day") +
  ylab("Service Hours Per Day") +
  scale_x_discrete(name = "Route Number") +
  scale_fill_phl(palette = "main", discrete = T) +
  labs(fill= "", title = paste0("The Corridor Serves ", round(sum(route_analytics$service_hours, na.rm =
  #theme_phl() +
  theme(legend.position = "top")+
  theme(text = element_text(size = 9))
```

The Corridor Serves 85.7 Service Hours per Day
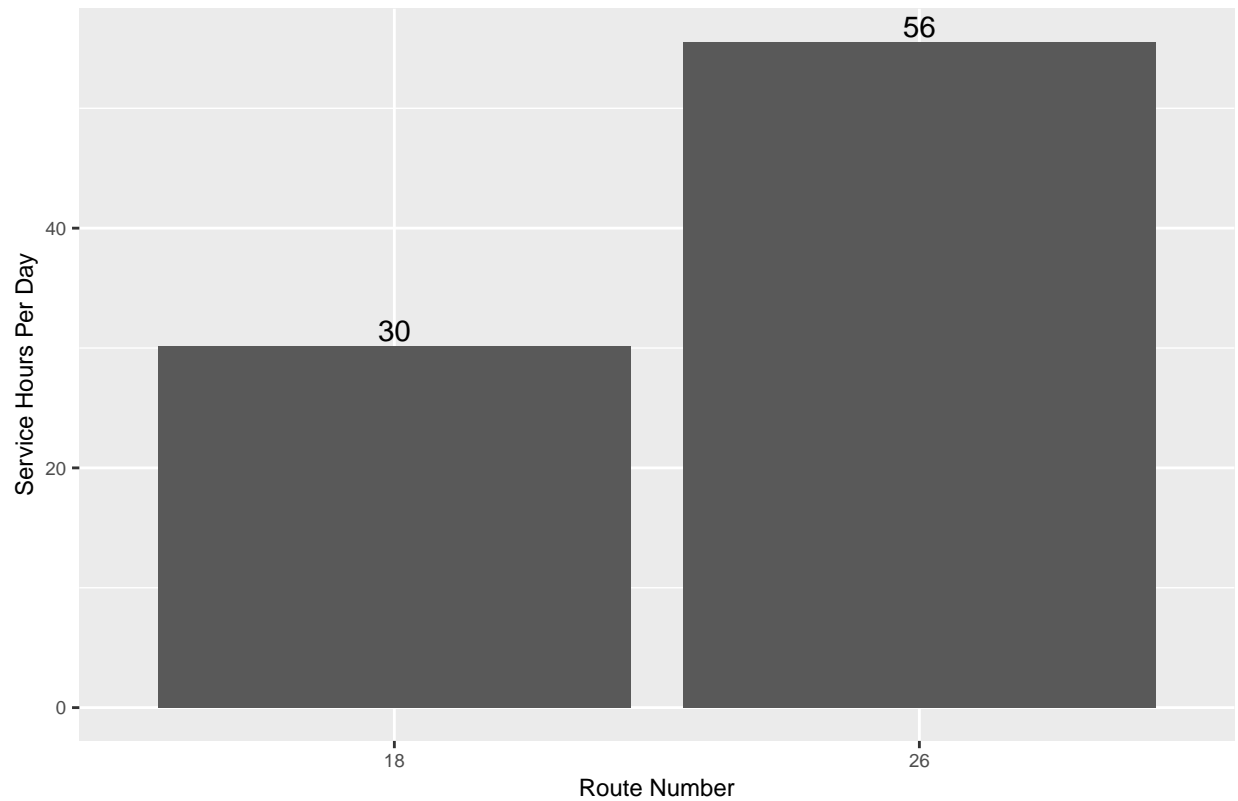


```
ggplot(hourly_route_analytics, aes(x = trip_hour, y = daily_ridership, group=route_id, fill=route_id))
  geom_bar(position = "stack", stat = "identity") +
  #geom_point(shape=21, size=3) +
  scale_color_viridis(discrete = TRUE) +
  scale_y_continuous(name = "Riders Served", n.breaks = 8) +
  scale_x_continuous(name = "Hour", breaks = c(0:23)) +
  labs(title = "Hourly Ridership by Route (all directions)") +
  theme(text = element_text(size = 9))
```
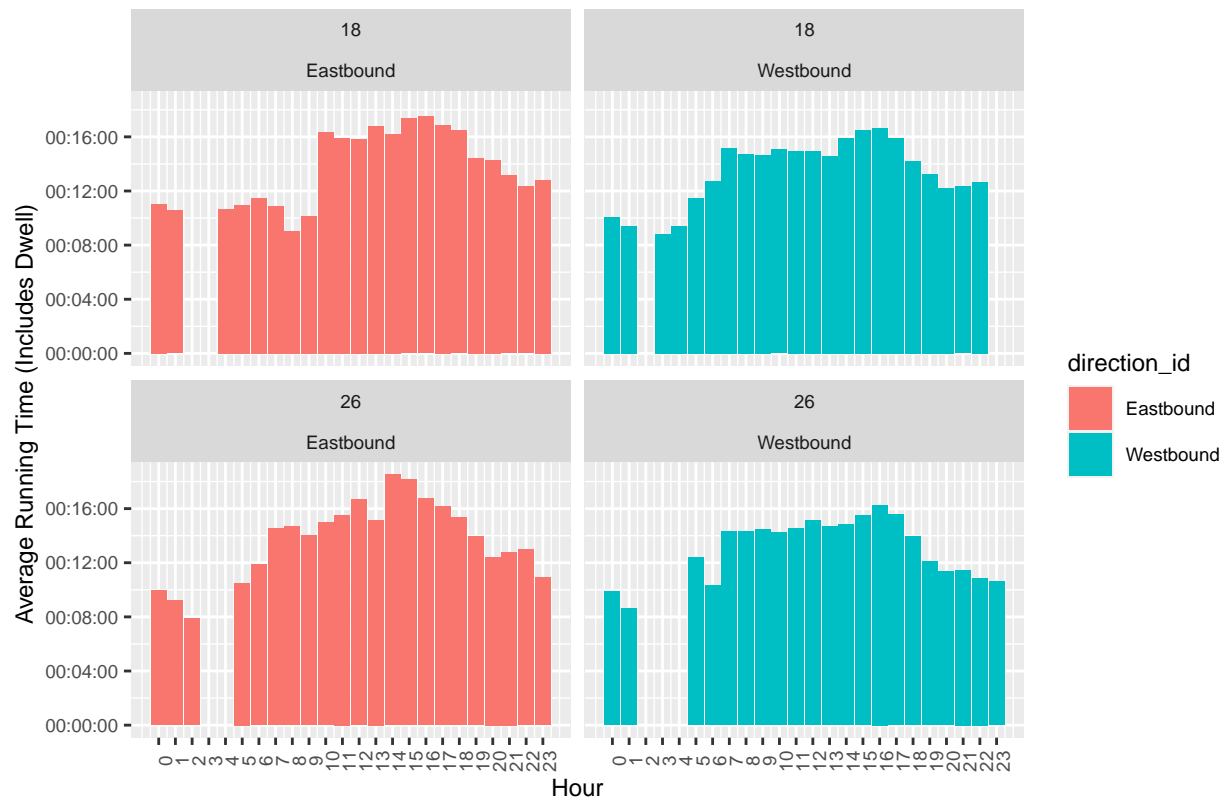
Hourly Ridership by Route (all directions)
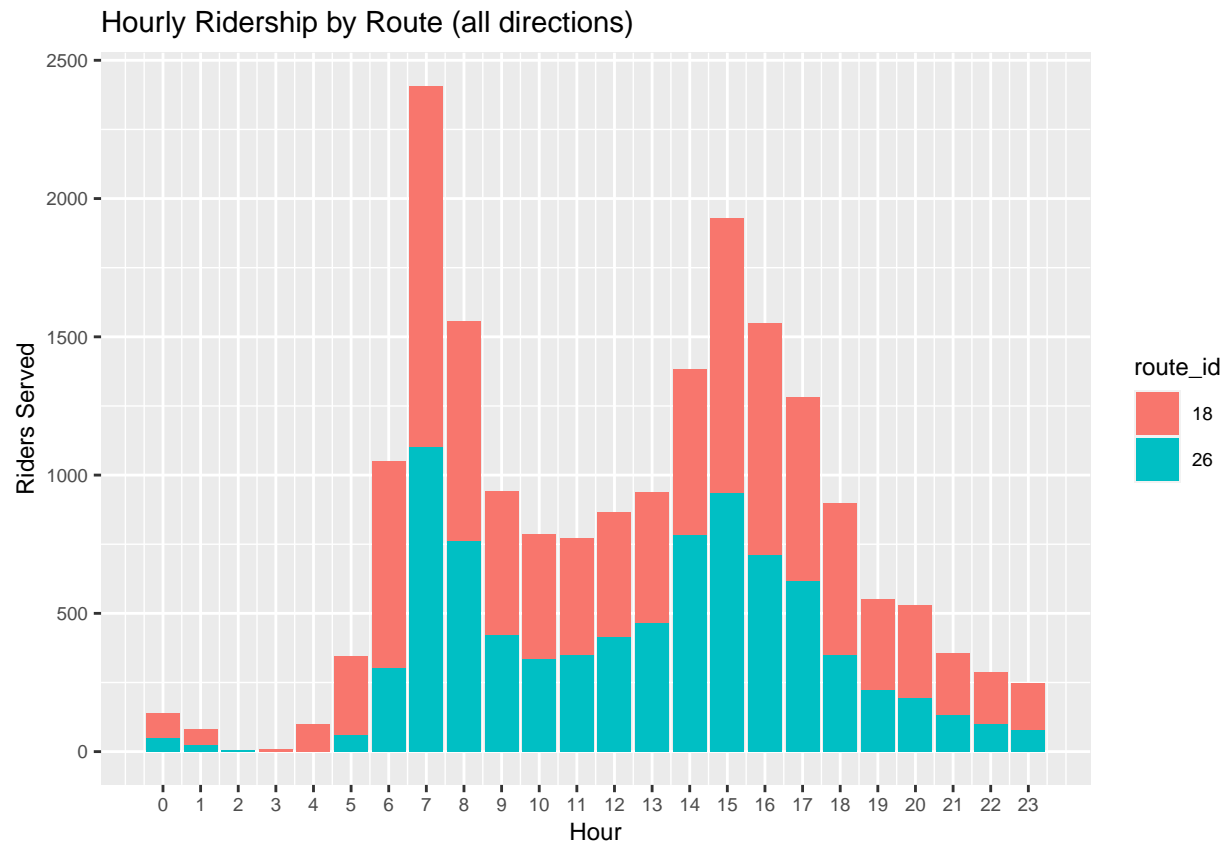

```
ridership_by_route_dir_plot(hourly_route_direction_analytics)
```

Hourly Ridership by Route/Direction
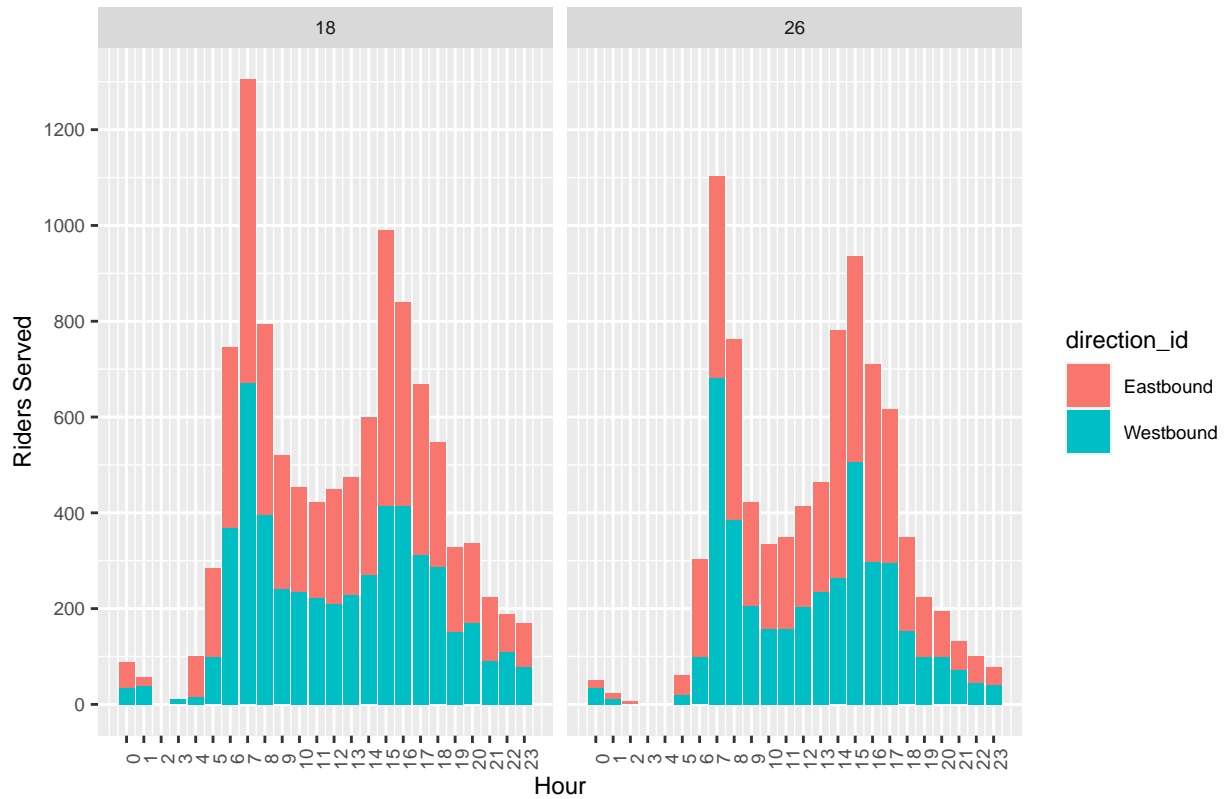


```
ggplot(hourly_route_direction_analytics %>% subset(avg_run >=0), aes(x = trip_hour, y = avg_run, fill=di
  geom_bar(position = "stack", stat = "identity") +
  #geom_point(size=3) +
  #scale_fill_viridis(discrete = TRUE) +
  ylab("Average Running Time (Includes Dwell)") +
  scale_x_continuous(name = "Hour", breaks = c(0:23)) +
  labs(title = "Average Corridor Running Time by Route/Direction") +
  theme(text = element_text(size = 9)) +
  facet_wrap(~route_id + direction_id, ncol = 2) +
  #theme_phl(base_size = 9)
  theme(text = element_text(size = 9), axis.text.x = element_text(angle = 90))
```

Average ~~Corridor~~ Running Time by Route/Direction


```
speed_by_route_dir_plot(hourly_route_direction_analytics)
```

Hourly Speed by Route/Direction
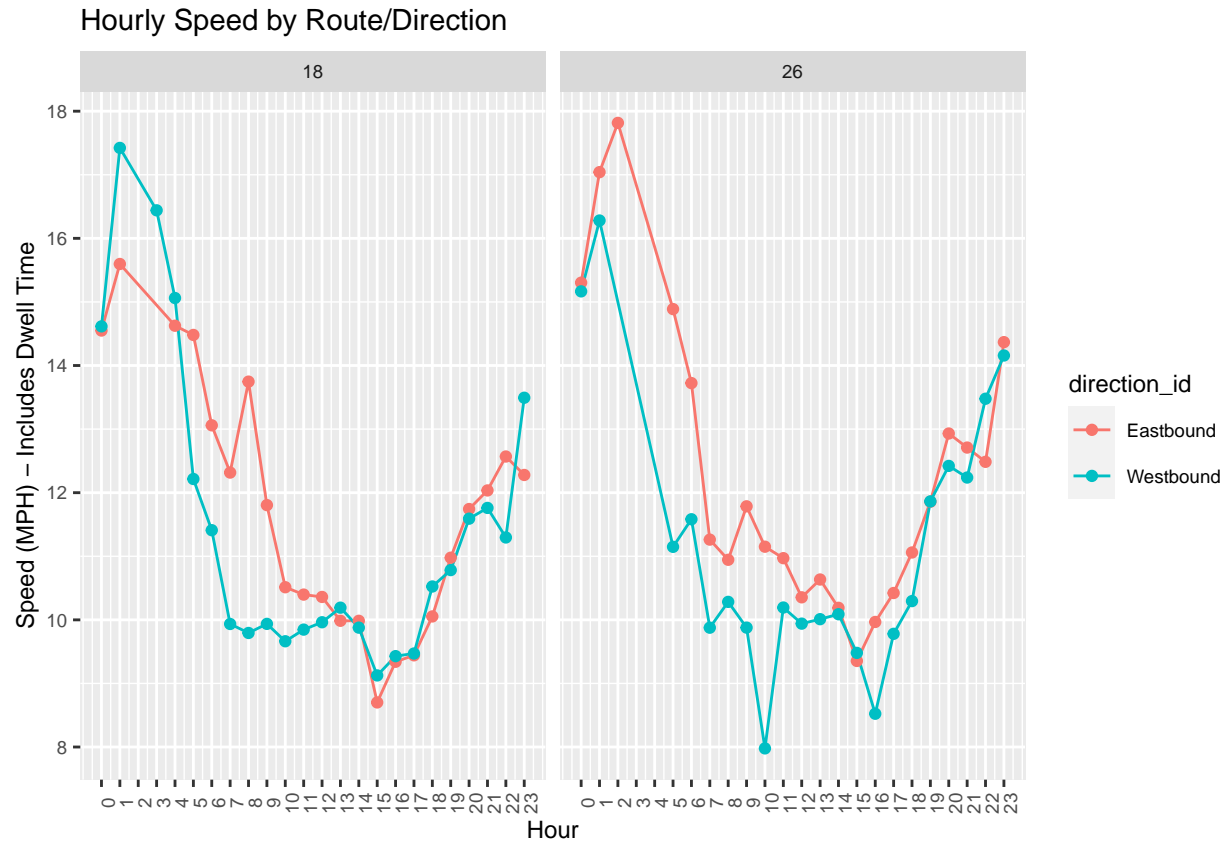
Note: average speed and running time are calculated for the entire running time, including dwell times.

Sub-Corridor Analysis

There are three main "sub-corridors" within the Olney corridor: Chelten to Broad, which is the section west of Broad St, Broad to 7th, and 7th to Front, which are both east of Broad St.

Chelten to Broad

```
stop_cb <- c("SEPTA372","SEPTA15911","SEPTA15794", "SEPTA15912", "SEPTA15789", "SEPTA15915", "SEPTA1558

trip_dat_cb <- (find_trip_dat_v2(apc_data, stop_cb))
```

```
## [1] "Running passenger data for 462 trips"
```

```
analytics_cb <-    analyze_segment(trip_dat_cb)
route_analytics_cb <- analyze_segment_route(trip_dat_cb)
binned_analytics_cb <- suppressWarnings(analyze_segment_hourbin(trip_dat_cb))
hourly_analytics_cb <- suppressWarnings(analyze_segment_hourly(trip_dat_cb))
hourly_route_analytics_cb <- suppressWarnings(analyze_segment_route_hourly(trip_dat_cb))
hourly_route_direction_analytics_cb <- suppressWarnings(analyze_segment_route_direction_hourly(trip_dat_
```

```
p <- ggplot(route_analytics_cb, aes(y = daily_ridership, x = route_id, label = daily_ridership)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  geom_text(position=position_dodge(width=0.9), vjust = -0.75, size = 3) +
  scale_y_continuous(name = "Total Riders Served", n.breaks = 8, limits = c(0, max(route_analytics$daily
  scale_x_discrete(name = "Route Number") +
  labs(title = paste0("The Corridor Serves ", round(sum(analytics$daily_ridership)), " Riders per Day")
  theme(text = element_text(size = 9))

ggplot(hourly_analytics_cb) +
  geom_line(aes(x = trip_hour, y = avg_segment_speed)) +
  geom_point(aes(x = trip_hour, y = avg_segment_speed), shape=21, size=3) +
  scale_y_continuous(name = "Speed (MPH) - Includes Dwell Time", n.breaks = 8) +
  scale_x_continuous(name = "Hour", breaks = c(0:23)) +
  labs(title = "Hourly Speed (All Routes)")+
  theme(text = element_text(size = 9))
```
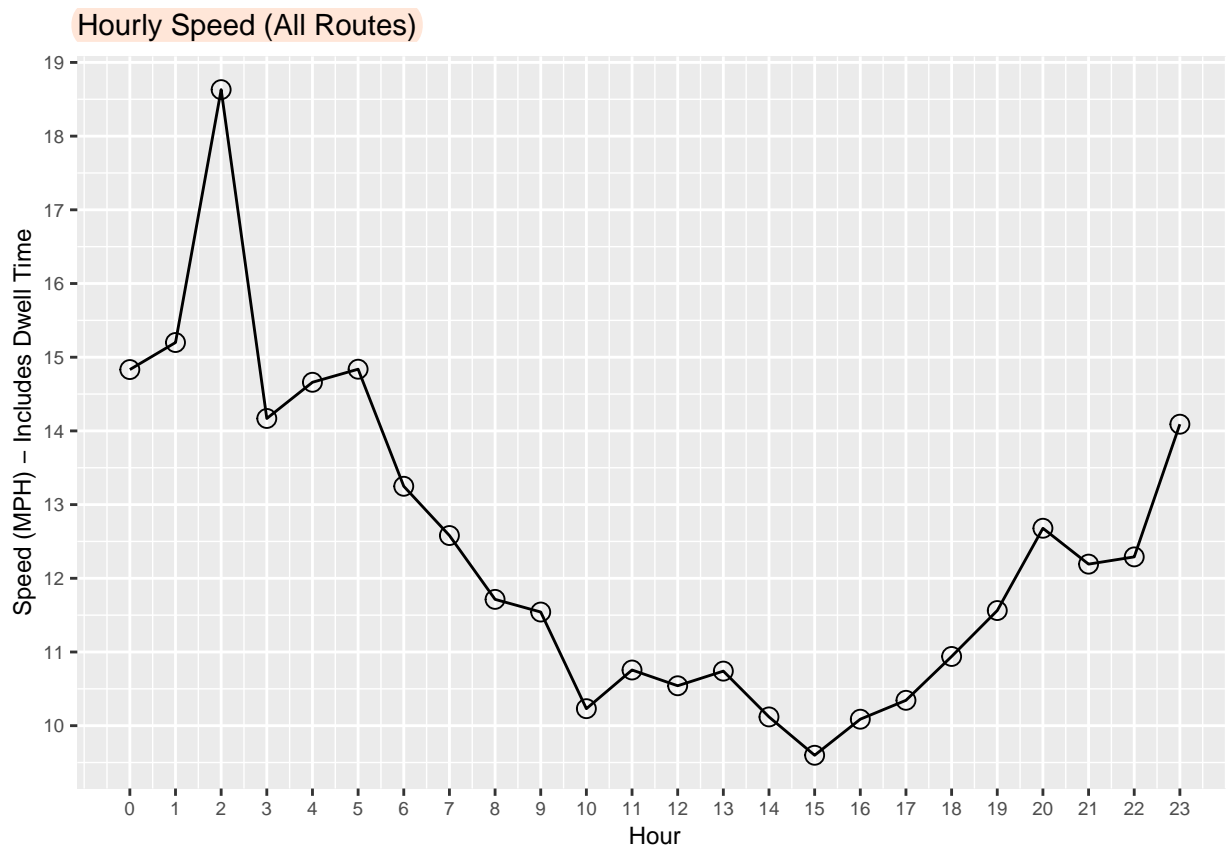


```
speed_by_route_dir_plot(hourly_route_direction_analytics_cb)
```

Hourly Speed by Route/Direction
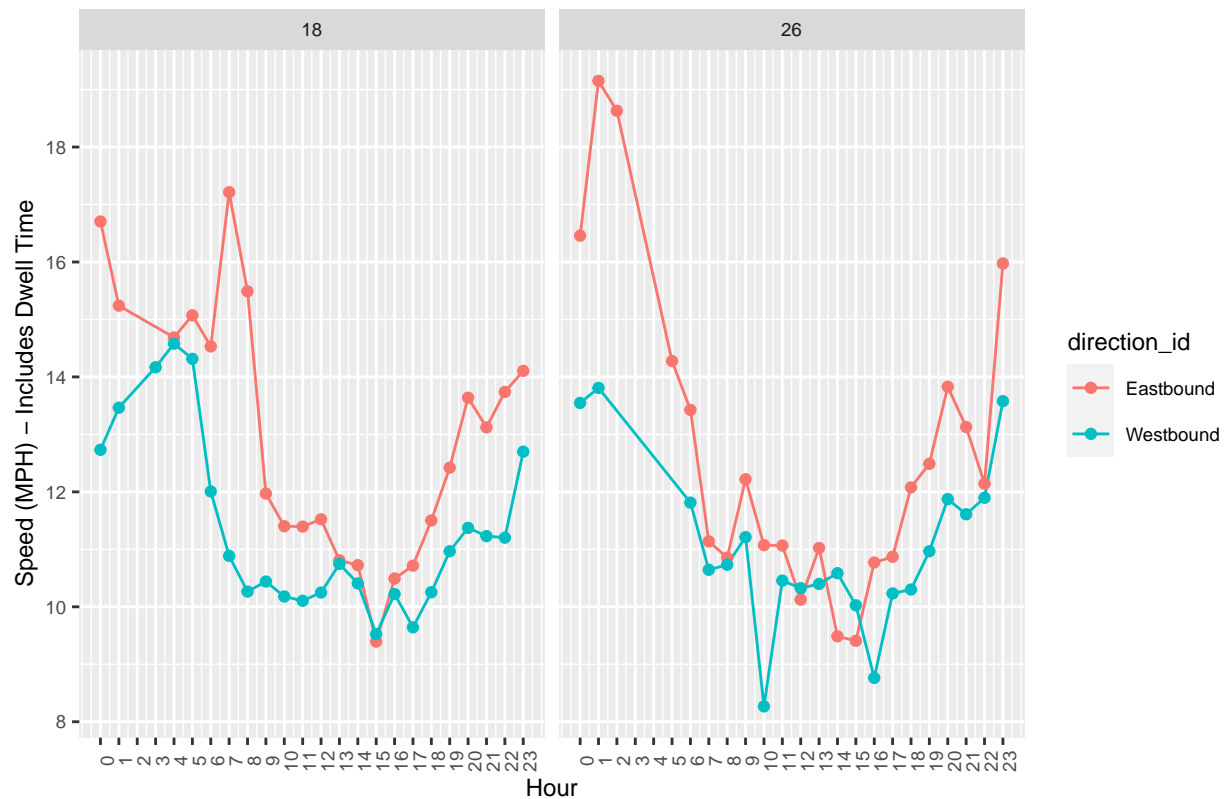
Broad to 7th

```
stop_b7 <- c("SEPTA15796", "SEPTA15908", "SEPTA15798", "SEPTA15907", "SEPTA15799", "SEPTA15800", "SEPTA

trip_dat_b7 <- (find_trip_dat_v2(apc_data, stop_b7))
```

```
## [1] "Running passenger data for 464 trips"
```

```
analytics_b7 <- analyze_segment(trip_dat_b7)
```

7th to Front

```
stop_7f <- c("SEPTA16963", "SEPTA15815", "SEPTA16964", "SEPTA15817","SEPTA15819", "SEPTA16961", "SEPTA3

trip_dat_7f <- (find_trip_dat_v2(apc_data, stop_7f))
```

```
## [1] "Running passenger data for 452 trips"
```

```
analytics_7f <- analyze_segment(trip_dat_7f)
```