Accident coursework

```
library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse
1.3.1 ──

## ✓ ggplot2 3.3.5     ✓ purrr   0.3.4
## ✓ tibble  3.1.5     ✓ dplyr   1.0.7
## ✓ tidyr   1.1.4     ✓ stringr 1.4.0
## ✓ readr   2.0.2     ✓ forcats 0.5.1

## ── Conflicts ──────────────────────────────────────────
tidyverse_conflicts() ──
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

library(ggplot2)
library(semTools)

## Loading required package: lavaan

## This is lavaan 0.6-9
## lavaan is FREE software! Please report any bugs.

## 

## 
##############################################################################
## 

## This is semTools 0.5-5

## All users of R (or SEM) are invited to submit functions or ideas for
functions.

## 
##############################################################################
## 

## 
## Attaching package: 'semTools'

## The following object is masked from 'package:readr':
## 
##     clipboard

library(waffle)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union

library(dplyr)
library(scales)

##
## Attaching package: 'scales'

## The following object is masked from 'package:purrr':
##
##     discard

## The following object is masked from 'package:readr':
##
##     col_factor

library(descr)
```

Recoding the categorical variables to clean the data

```
#-----------------------Data Cleaning---------------------------------

Accident<-read.csv("/Users/komal/OneDrive - University of
Southampton/Postgraduate/E-books/BAMS/Foundations of Business Analytics and
Management/Coursework/Accidents.csv")
Accident.cleaned<-Accident%>%
  mutate(Police_Force= recode_factor(.x=Police_Force,"1"="Metropolitan
Police",
"3"="Cumbria",
"4"="Lancashire",
"5"="Merseyside",
"6"="Greater Manchester",
"7"="Cheshire",
"10"="Northumbria",
"11"="Durham",
"12"="North Yorkshire",
"13"="West Yorkshire",
"14"="South Yorkshire",
"16"="Humberside",
"17"="Cleveland",
"20"="West Midlands",
"21"="Staffordshire",
"22"="West Mercia",
"23"="Warwickshire",
"30"="Derbyshire",
"31"="Nottinghamshire",
```

```
"32"="Lincolnshire",
"33"="Leicestershire",
"34"="Northamptonshire",
"35"="Cambridgeshire",
"36"="Norfolk",
"37"="Suffolk",
"40"="Bedfordshire",
"41"="Hertfordshire",
"42"="Essex",
"43"="Thames Valley",
"44"="Hampshire",
"45"="Surrey",
"46"="Kent",
"47"="Sussex",
"48"="City of London",
"50"="Devon and Cornwall",
"52"="Avon and Somerset",
"53"="Gloucestershire",
"54"="Wiltshire",
"55"="Dorset",
"60"="North Wales",
"61"="Gwent",
"62"="South Wales",
"63"="Dyfed-Powys",
"91"="Northern",
"92"="Grampian",
"93"="Tayside",
"94"="Fife",
"95"="Lothian and Borders",
"96"="Central",
"97"="Strathclyde",
"98"="Dumfries and Galloway"))%>%
  mutate(Accident_Severity=recode_factor(.x=Accident_Severity,"1"="Fatal",
"2"="Serious",
"3"="Slight"))%>%
  mutate(Day_of_Week=recode_factor(.x=Day_of_Week,"1"="Sunday",
"2"="Monday",
"3"="Tuesday",
"4"="Wednesday",
"5"="Thursday",
"6"="Friday",
"7"="Saturday"))%>%

mutate(Local_Authority_District=recode_factor(.x=Local_Authority_District,"1"
="Westminster",
"2"="Camden",
"3"="Islington",
"4"="Hackney",
"5"="Tower Hamlets",
"6"="Greenwich",
```

```
"7"="Lewisham",
"8"="Southwark",
"9"="Lambeth",
"10"="Wandsworth",
"11"="Hammersmith and Fulham",
"12"="Kensington and Chelsea",
"13"="Waltham Forest",
"14"="Redbridge",
"15"="Havering",
"16"="Barking and Dagenham",
"17"="Newham",
"18"="Bexley",
"19"="Bromley",
"20"="Croydon",
"21"="Sutton",
"22"="Merton",
"23"="Kingston upon Thames",
"24"="Richmond upon Thames",
"25"="Hounslow",
"26"="Hillingdon",
"27"="Ealing",
"28"="Brent",
"29"="Harrow",
"30"="Barnet",
"31"="Haringey",
"32"="Enfield",
"33"="Hertsmere",
"38"="Epsom and Ewell",
"40"="Spelthorne",
"57"="London Airport (Heathrow)",
"60"="Allerdale",
"61"="Barrow-in-Furness",
"62"="Carlisle",
"63"="Copeland",
"64"="Eden",
"65"="South Lakeland",
"70"="Blackburn with Darwen",
"71"="Blackpool",
"72"="Burnley",
"73"="Chorley",
"74"="Fylde",
"75"="Hyndburn",
"76"="Lancaster",
"77"="Pendle",
"79"="Preston",
"80"="Ribble Valley",
"82"="Rossendale",
"83"="South Ribble",
"84"="West Lancashire",
"85"="Wyre",
```

```
"90"="Knowsley",
"91"="Liverpool",
"92"="St. Helens",
"93"="Sefton",
"95"="Wirral",
"100"="Bolton",
"101"="Bury",
"102"="Manchester",
"104"="Oldham",
"106"="Rochdale",
"107"="Salford",
"109"="Stockport",
"110"="Tameside",
"112"="Trafford",
"114"="Wigan",
"120"="Chester",
"121"="Congleton",
"122"="Crewe and Nantwich",
"123"="Ellesmere Port and Neston",
"124"="Halton",
"126"="Macclesfield",
"127"="Vale Royal",
"128"="Warrington",
"129"="Cheshire East",
"130"="Cheshire West and Chester",
"139"="Northumberland",
"140"="Alnwick",
"141"="Berwick-upon-Tweed",
"142"="Blyth Valley",
"143"="Castle Morpeth",
"144"="Tynedale",
"145"="Wansbeck",
"146"="Gateshead",
"147"="Newcastle upon Tyne",
"148"="North Tyneside",
"149"="South Tyneside",
"150"="Sunderland",
"160"="Chester-le-Street",
"161"="Darlington",
"162"="Derwentside",
"163"="Durham",
"164"="Easington",
"165"="Sedgefield",
"166"="Teesdale",
"168"="Wear Valley",
"169"="County Durham",
"180"="Craven",
"181"="Hambleton",
"182"="Harrogate",
"184"="Richmondshire",
```

```
"185"="Ryedale",
"186"="Scarborough",
"187"="Selby",
"189"="York",
"200"="Bradford",
"202"="Calderdale",
"203"="Kirklees",
"204"="Leeds",
"206"="Wakefield",
"210"="Barnsley",
"211"="Doncaster",
"213"="Rotherham",
"215"="Sheffield",
"228"="Kingston upon Hull, City of",
"231"="East Riding of Yorkshire",
"232"="North Lincolnshire",
"233"="North East Lincolnshire",
"240"="Hartlepool",
"241"="Redcar and Cleveland",
"243"="Middlesbrough",
"245"="Stockton-on-Tees",
"250"="Cannock Chase",
"251"="East Staffordshire",
"252"="Lichfield",
"253"="Newcastle-under-Lyme",
"254"="South Staffordshire",
"255"="Stafford",
"256"="Staffordshire Moorlands",
"257"="Stoke-on-Trent",
"258"="Tamworth",
"270"="Bromsgrove",
"273"="Malvern Hills",
"274"="Redditch",
"276"="Worcester",
"277"="Wychavon",
"278"="Wyre Forest",
"279"="Bridgnorth",
"280"="North Shropshire",
"281"="Oswestry",
"282"="Shrewsbury and Atcham",
"283"="South Shropshire",
"284"="Telford and Wrekin",
"285"="Herefordshire, County of ",
"286"="Shropshire",
"290"="North Warwickshire",
"291"="Nuneaton and Bedworth",
"292"="Rugby ",
"293"="Stratford-upon-Avon",
"294"="Warwick",
"300"="Birmingham",
```

```
"302"="Coventry",
"303"="Dudley",
"305"="Sandwell",
"306"="Solihull",
"307"="Walsall",
"309"="Wolverhampton",
"320"="Amber Valley",
"321"="Bolsover",
"322"="Chesterfield",
"323"="Derby",
"324"="Erewash",
"325"="High Peak",
"327"="North East Derbyshire",
"328"="South Derbyshire",
"329"="Derbyshire Dales",
"340"="Ashfield",
"341"="Bassetlaw",
"342"="Broxtowe",
"343"="Gedling",
"344"="Mansfield",
"345"="Newark and Sherwood",
"346"="Nottingham",
"347"="Rushcliffe",
"350"="Boston",
"351"="East Lindsey",
"352"="Lincoln",
"353"="North Kesteven",
"354"="South Holland",
"355"="South Kesteven",
"356"="West Lindsey",
"360"="Blaby",
"361"="Hinckley and Bosworth",
"362"="Charnwood",
"363"="Harborough",
"364"="Leicester",
"365"="Melton",
"366"="North West Leicestershire",
"367"="Oadby and Wigston",
"368"="Rutland",
"380"="Corby",
"381"="Daventry",
"382"="East Northamptonshire",
"383"="Kettering",
"384"="Northampton",
"385"="South Northamptonshire",
"386"="Wellingborough",
"390"="Cambridge",
"391"="East Cambridgeshire",
"392"="Fenland",
"393"="Huntingdonshire",
```

```
"394"="Peterborough",
"395"="South Cambridgeshire",
"400"="Breckland",
"401"="Broadland",
"402"="Great Yarmouth",
"404"="Norwich",
"405"="North Norfolk",
"406"="South Norfolk",
"407"="King's Lynn and West Norfolk",
"410"="Babergh",
"411"="Forest Heath",
"412"="Ipswich",
"413"="Mid Suffolk",
"414"="St. Edmundsbury",
"415"="Suffolk Coastal",
"416"="Waveney",
"420"="Bedford",
"421"="Luton",
"422"="Mid Bedfordshire",
"423"="South Bedfordshire",
"424"="Central Bedfordshire",
"430"="Broxbourne",
"431"="Dacorum",
"432"="East Hertfordshire",
"433"="North Hertfordshire",
"434"="St. Albans",
"435"="Stevenage",
"436"="Three Rivers",
"437"="Watford",
"438"="Welwyn Hatfield",
"450"="Basildon",
"451"="Braintree",
"452"="Brentwood",
"453"="Castle Point",
"454"="Chelmsford",
"455"="Colchester",
"456"="Epping Forest",
"457"="Harlow",
"458"="Maldon",
"459"="Rochford",
"460"="Southend-on-Sea",
"461"="Tendring",
"462"="Thurrock",
"463"="Uttlesford",
"470"="Bracknell Forest",
"471"="West Berkshire",
"472"="Reading",
"473"="Slough",
"474"="Windsor and Maidenhead",
"475"="Wokingham",
```

```
"476"="Aylesbury Vale",
"477"="South Bucks",
"478"="Chiltern",
"479"="Milton Keynes",
"480"="Wycombe",
"481"="Cherwell",
"482"="Oxford",
"483"="Vale of White Horse",
"484"="South Oxfordshire",
"485"="West Oxfordshire",
"490"="Basingstoke and Deane",
"491"="Eastleigh",
"492"="Fareham",
"493"="Gosport",
"494"="Hart",
"495"="Havant",
"496"="New Forest",
"497"="East Hampshire",
"498"="Portsmouth",
"499"="Rushmoor",
"500"="Southampton ",
"501"="Test Valley",
"502"="Winchester",
"505"="Isle of Wight",
"510"="Elmbridge",
"511"="Guildford",
"512"="Mole Valley",
"513"="Reigate and Banstead",
"514"="Runnymede",
"515"="Surrey Heath",
"516"="Tandridge",
"517"="Waverley",
"518"="Woking",
"530"="Ashford",
"531"="Canterbury",
"532"="Dartford",
"533"="Dover",
"535"="Gravesham",
"536"="Maidstone",
"538"="Sevenoaks",
"539"="Shepway",
"540"="Swale",
"541"="Thanet",
"542"="Tonbridge and Malling",
"543"="Tunbridge Wells",
"544"="Medway",
"551"="Eastbourne",
"552"="Hastings",
"554"="Lewes",
"555"="Rother",
```

```
"556"="Wealden",
"557"="Adur",
"558"="Arun",
"559"="Chichester",
"560"="Crawley",
"562"="Horsham",
"563"="Mid Sussex",
"564"="Worthing",
"565"="Brighton and Hove",
"570"="City of London",
"580"="East Devon",
"581"="Exeter",
"582"="North Devon",
"583"="Plymouth",
"584"="South Hams",
"585"="Teignbridge",
"586"="Mid Devon",
"587"="Torbay",
"588"="Torridge",
"589"="West Devon",
"590"="Caradon",
"591"="Carrick",
"592"="Kerrier",
"593"="North Cornwall",
"594"="Penwith",
"595"="Restormel",
"596"="Cornwall",
"601"="Bristol, City of",
"605"="North Somerset",
"606"="Mendip",
"607"="Sedgemoor",
"608"="Taunton Deane",
"609"="West Somerset",
"610"="South Somerset",
"611"="Bath and North East Somerset",
"612"="South Gloucestershire",
"620"="Cheltenham",
"621"="Cotswold",
"622"="Forest of Dean",
"623"="Gloucester",
"624"="Stroud",
"625"="Tewkesbury",
"630"="Kennet",
"631"="North Wiltshire",
"632"="Salisbury",
"633"="Swindon",
"634"="West Wiltshire",
"635"="Wiltshire",
"640"="Bournemouth",
"641"="Christchurch",
```

```
"642"="North Dorset",
"643"="Poole",
"644"="Purbeck",
"645"="West Dorset",
"646"="Weymouth and Portland",
"647"="East Dorset",
"720"="Isle of Anglesey",
"721"="Conwy",
"722"="Gwynedd",
"723"="Denbighshire",
"724"="Flintshire",
"725"="Wrexham",
"730"="Blaenau Gwent",
"731"="Caerphilly",
"732"="Monmouthshire",
"733"="Newport",
"734"="Torfaen",
"740"="Bridgend",
"741"="Cardiff",
"742"="Merthyr Tydfil",
"743"="Neath Port Talbot",
"744"="Rhondda, Cynon, Taff",
"745"="Swansea",
"746"="The Vale of Glamorgan",
"750"="Ceredigion",
"751"="Carmarthenshire",
"752"="Pembrokeshire",
"753"="Powys",
"910"="Aberdeen City",
"911"="Aberdeenshire",
"912"="Angus",
"913"="Argyll and Bute",
"914"="Scottish Borders",
"915"="Clackmannanshire",
"916"="West Dunbartonshire",
"917"="Dumfries and Galloway",
"918"="Dundee City",
"919"="East Ayrshire",
"920"="East Dunbartonshire",
"921"="East Lothian",
"922"="East Renfrewshire",
"923"="Edinburgh, City of",
"924"="Falkirk",
"925"="Fife",
"926"="Glasgow City",
"927"="Highland",
"928"="Inverclyde",
"929"="Midlothian",
"930"="Moray",
"931"="North Ayrshire",
```

```
"932"="North Lanarkshire",
"933"="Orkney Islands",
"934"="Perth and Kinross",
"935"="Renfrewshire",
"936"="Shetland Islands",
"937"="South Ayrshire",
"938"="South Lanarkshire",
"939"="Stirling",
"940"="West Lothian",
"941"="Western Isles"))%>%
  mutate(Road_Class_1st=recode_factor(.x=Road_Class_1st,"1"="Motorway",
"2"="A(M)",
"3"="A",
"4"="B",
"5"="C",
"6"="Unclassified"))%>%
  mutate(Road_Type=recode_factor(.x=Road_Type,"1"="Roundabout",
"2"="One way street",
"3"="Dual carriageway",
"6"="Single carriageway",
"7"="Slip road",
"9"=NA_character_,
"12"="One way street/Slip road",
"-1"=NA_character_))%>%
  mutate(Junction_Detail=recode_factor(.x=Junction_Detail,"0"="Not at
junction or within 20 metres",
"1"="Roundabout",
"2"="Mini-roundabout",
"3"="T or staggered junction",
"5"="Slip road",
"6"="Crossroads",
"7"="More than 4 arms (not roundabout)",
"8"="Private drive or entrance",
"9"="Other junction",
"-1"=NA_character_))%>%
  mutate(Junction_Control=recode_factor(.x=Junction_Control,"0"="Not at
junction or within 20 metres",
"1"="Authorised person",
"2"="Auto traffic signal",
"3"="Stop sign",
"4"="Give way or uncontrolled",
"-1"=NA_character_))%>%
  mutate(Road_Class_2nd=recode_factor(.x=Road_Class_2nd,"0"="Not at junction
or within 20 metres",
"1"="Motorway",
"2"="A(M)",
"3"="A",
"4"="B",
"5"="C",
"6"="Unclassified"))%>%
```

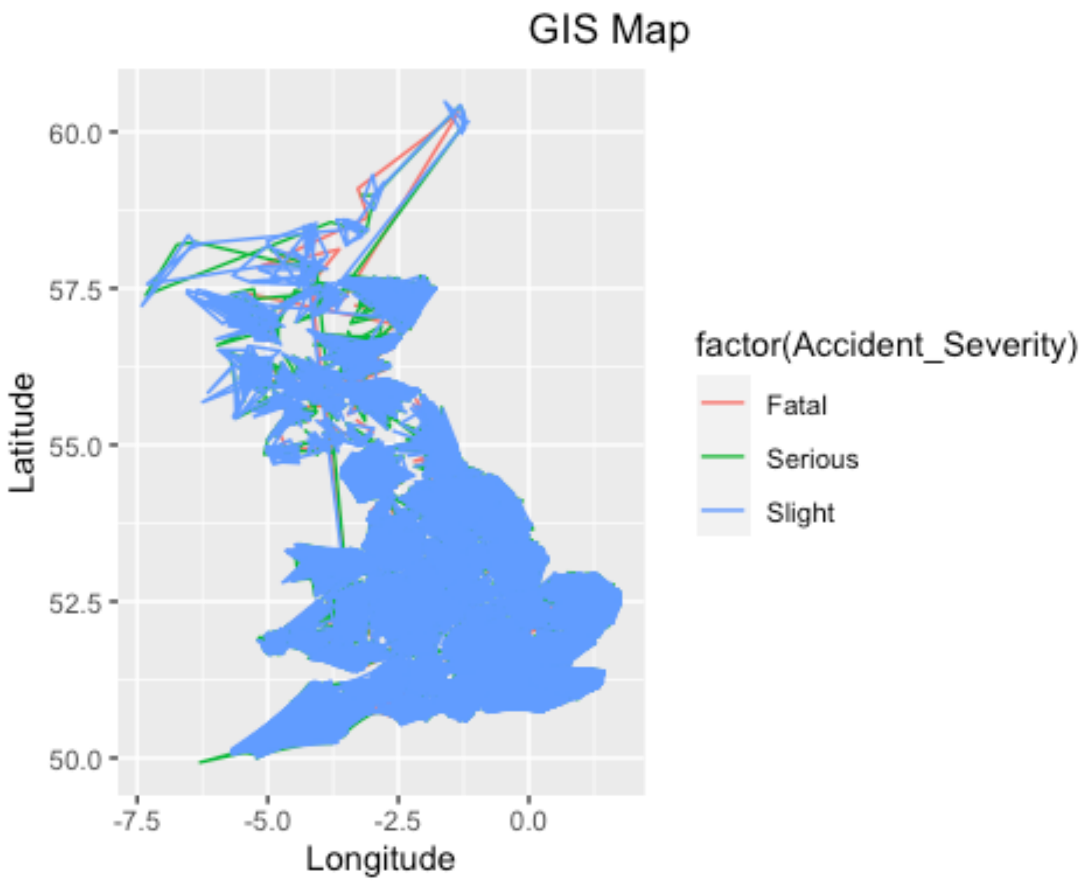
```
mutate(Pedestrian_Crossing_Human_Control=recode_factor(.x=Pedestrian_Crossing
_Human_Control,"0"="None within 50 metres ",
"1"="Control by school crossing patrol",
"2"="Control by other authorised person",
"-1"=NA_character_))%>%

mutate(Pedestrian_Crossing_Physical_Facilities=recode_factor(.x=Pedestrian_Cr
ossing_Physical_Facilities,"0"="No physical crossing facilities within 50
metres",
"1"="Zebra",
"4"="Pelican, puffin, toucan or similar non-junction pedestrian light
crossing",
"5"="Pedestrian phase at traffic signal junction",
"7"="Footbridge or subway",
"8"="Central refuge",
"-1"="No physical crossing facilities within 50 metres"))%>%
  mutate(Light_Conditions=recode_factor(.x = Light_Conditions,"1"="Daylight",
"4"="Darkness - lights lit",
"5"="Darkness - lights unlit",
"6"="Darkness - no lighting",
"7"="Darkness - lighting unknown",
"-1"=NA_character_))%>%
  mutate(Weather_Conditions=recode_factor(.x = Weather_Conditions,"1"="Fine
no high winds",
"2"="Raining no high winds",
"3"="Snowing no high winds",
"4"="Fine + high winds",
"5"="Raining + high winds",
"6"="Snowing + high winds",
"7"="Fog or mist",
"8"="Fine no high winds",
"9"="Fine no high winds",
"-1"="Fine no high winds"))%>%
  mutate(Road_Surface_Conditions=recode_factor(.x =
Road_Surface_Conditions,"1"="Dry",
"2"="Wet or damp",
"3"="Snow",
"4"="Frost or ice",
"5"="Flood over 3cm. deep",
"6"="Oil or diesel",
"7"="Mud",
"-1"=NA_character_))%>%
  mutate(Special_Conditions_at_Site=recode_factor(.x =
Special_Conditions_at_Site,"0"="None",
"1"="Auto traffic signal - out",
"2"="Auto signal part defective",
"3"="Road sign or marking defective or obscured",
"4"="Roadworks",
"5"="Road surface defective",
```

```
"6"="Oil or diesel",
"7"="Mud",
"-1"=NA_character_))%>%
  mutate(Carriageway_Hazards=recode_factor(.x =
Carriageway_Hazards,"0"="None",
"1"="Vehicle load on road",
"2"="Other object on road",
"3"="Previous accident",
"4"="Dog on road",
"5"="Other animal on road",
"6"="Pedestrian in carriageway - not injured",
"7"="Any animal in carriageway (except ridden horse)",
"-1"=NA_character_))%>%
  mutate(Urban_or_Rural_Area=recode_factor(.x =
Urban_or_Rural_Area,"1"="Urban",
"2"="Rural",
"3"="Unallocated"))%>%
  mutate(Did_Police_Officer_Attend_Scene_of_Accident=recode_factor(.x =
Did_Police_Officer_Attend_Scene_of_Accident,"1"="Yes",
"2"="No",
"3"="No - accident was reported using a self completion  form (self rep
only)"))%>%
  mutate(Latitude=as.numeric(x = Latitude))%>%
  mutate(Longitude=as.numeric(x = Longitude))
```

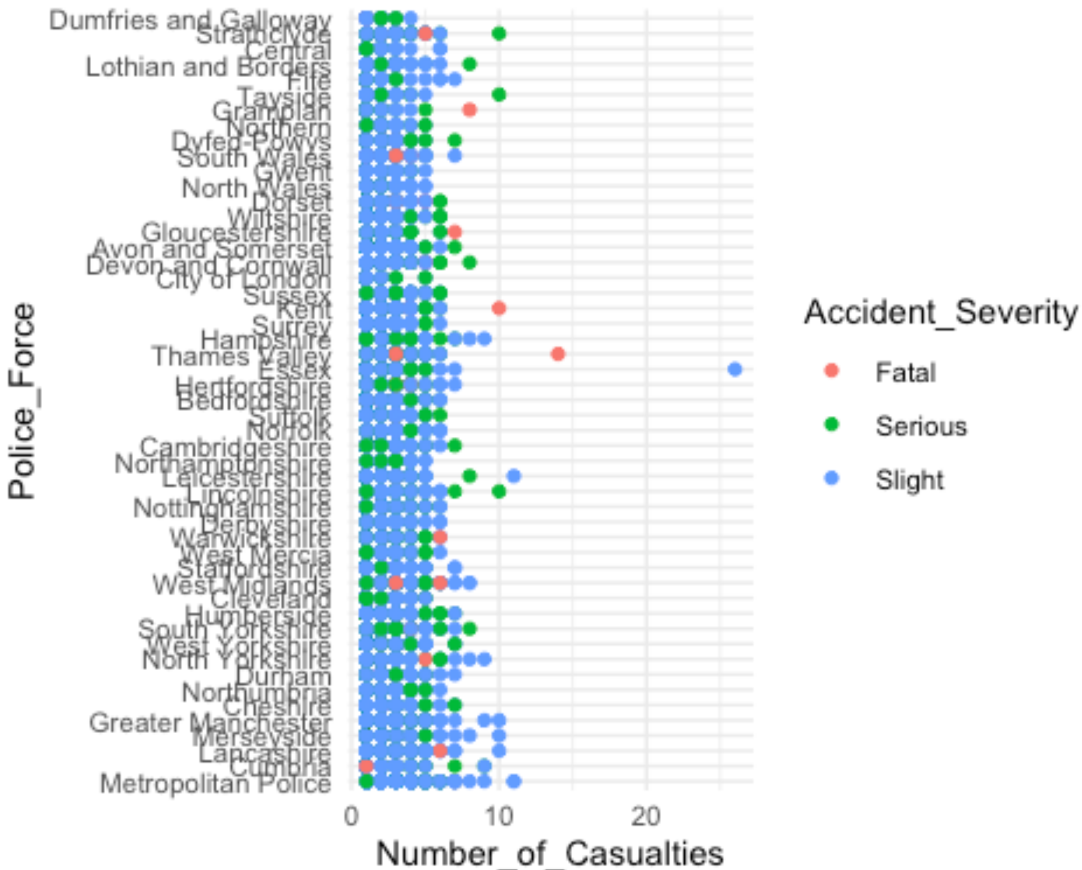
```
## Warning: Unreplaced values treated as NA as .x is not compatible. Please
specify
## replacements exhaustively or supply .default

## Warning in mask$eval_all_mutate(quo): NAs introduced by coercion

## Warning in mask$eval_all_mutate(quo): NAs introduced by coercion
```

Preliminary data analysis

```
#-----------Preliminary data analysis-------------------------
ggplot() +geom_path(data = Accident.cleaned, aes(x = Longitude, y = Latitude,
color = factor(Accident_Severity)))+labs(title = "
GIS Map")     #Creating GIS map
```

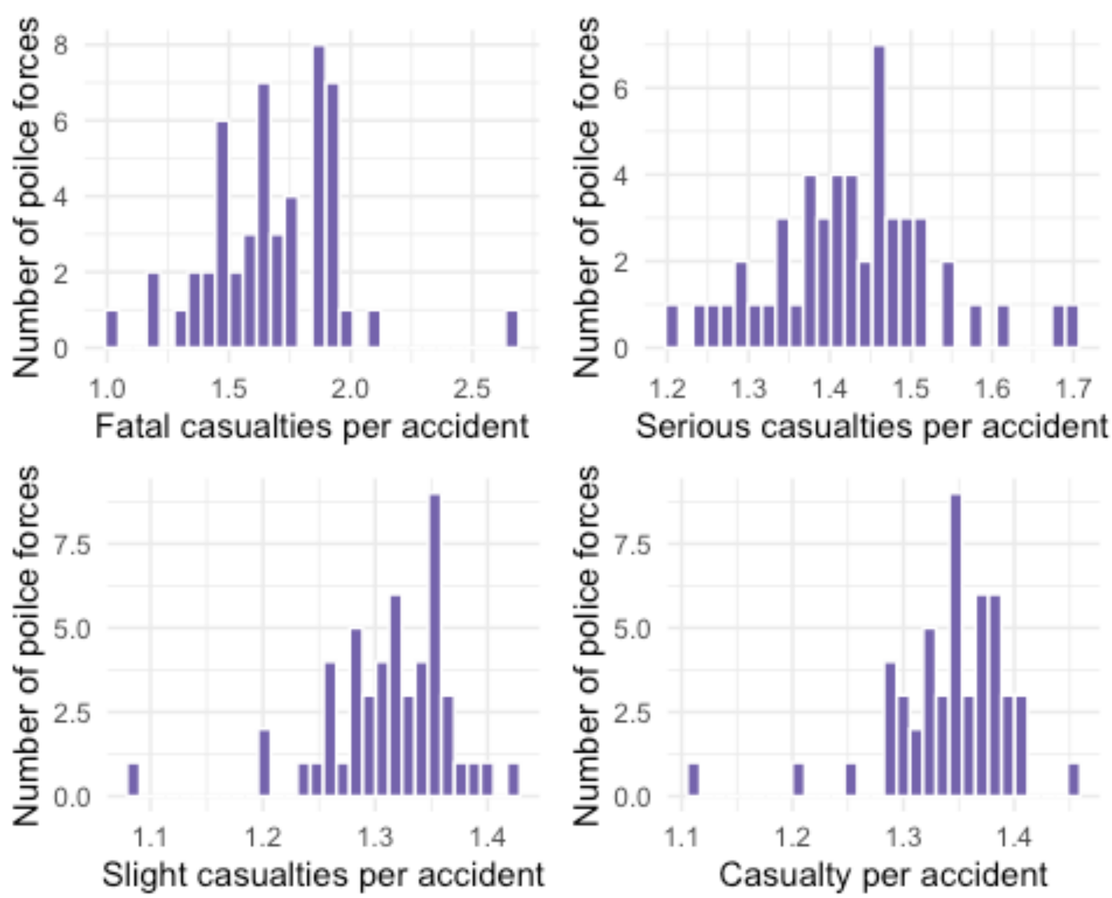



```
#--------------------Variation of accident severity-----------
Acc_severity<-Accident.cleaned%>%
  drop_na()%>%

ggplot(aes(x=Police_Force,y=Number_of_Casualties))+geom_point(aes(color=Accid
ent_Severity))+theme_minimal()+coord_flip()
Acc_severity
```




```
#---------------------Introducing casualty per accident--------
cas_per_acc<-Accident.cleaned%>%
  group_by(Police_Force)%>%
  summarise(cas.stat=mean(Number_of_Casualties))
police_severity<-Accident.cleaned%>%
  group_by(Police_Force,Accident_Severity)%>%
  summarise(cas.stat=mean(Number_of_Casualties))%>%
  spread(cas_per_acc,key = Accident_Severity,value = cas.stat)

## `summarise()` has grouped output by 'Police_Force'. You can override using
the `.groups` argument.

## Warning in if (!is.na(fill)) {: the condition has length > 1 and only the
first
## element will be used

police_severity$Fatal=as.numeric(police_severity$Fatal)
police_severity$Serious=as.numeric(police_severity$Serious)
police_severity$Slight=as.numeric(police_severity$Slight)
fatal.plot<-police_severity%>%

ggplot(aes(x=Fatal))+geom_histogram(fill="#7463AC",color="white")+theme_minim
al()+labs(x="Fatal casualties per accident", y="Number of poilce forces")
serious.plot<-police_severity%>%
```

```
ggplot(aes(x=Serious))+geom_histogram(fill="#7463AC",color="white")+theme_min
imal()+labs(x="Serious casualties per accident", y="Number of poilce forces")
  slight.plot<-police_severity%>%

ggplot(aes(x=Slight))+geom_histogram(fill="#7463AC",color="white")+theme_mini
mal()+labs(x="Slight casualties per accident", y="Number of poilce forces")
overall.plot<-ggplot(data=cas_per_acc,
aes(x=cas.stat))+geom_histogram(fill="#7463AC",color="white")+theme_minimal()
+labs(x="Casualty per accident",y="Number of police forces")
gridExtra::grid.arrange(fatal.plot,serious.plot,slight.plot,overall.plot,nrow
=2)

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
#--------------------------------Analysing normal distribution metrics------
--
cas_per_acc%>%
  summarise(mean.cas=mean(cas.stat),
            sd.cas=sd(cas.stat),
```

```
          var.cas=var(cas.stat),
          median.cas=median(cas.stat),
          skew.cas=skew(cas.stat),
          kurtosis.cas=kurtosis(cas.stat))

## # A tibble: 4 × 6
##   mean.cas sd.cas var.cas median.cas     skew.cas kurtosis.cas
##      <dbl>  <dbl>   <dbl>      <dbl>        <dbl>        <dbl>
## 1     1.34 0.0549 0.00302       1.35 -1.76               5.95
## 2     1.34 0.0549 0.00302       1.35  0.343              0.686
## 3     1.34 0.0549 0.00302       1.35 -5.12               8.67
## 4     1.34 0.0549 0.00302       1.35  0.000000308        0

mean.fatal=mean(police_severity$Fatal)
sd.fatal=sd(police_severity$Fatal)
skew.fatal=skew(police_severity$Fatal)
kurtosis.fatal=kurtosis(police_severity$Fatal)
mean.serious=mean(police_severity$Serious)
sd.serious=sd(police_severity$Serious)
skew.serious=skew(police_severity$Serious)
kurtosis.serious=kurtosis(police_severity$Serious)
mean.slight=mean(police_severity$Slight)
sd.slight=sd(police_severity$Slight)
skew.slight=skew(police_severity$Slight)
kurtosis.slight=kurtosis(police_severity$Slight)

skew.fatal

## skew (g1)        se         z         p
## 0.4198722 0.3429972 1.2241272 0.2209043

skew.serious

## skew (g1)        se         z         p
## 0.3010081 0.3429972 0.8775819 0.3801707

skew.slight

##     skew (g1)            se             z             p
## -1.438620e+00  3.429972e-01 -4.194262e+00  2.737613e-05

kurtosis.fatal

## Excess Kur (g2)              se               z               p
##    2.3385613722    0.6859943406    3.4090097161    0.0006519916

kurtosis.serious

## Excess Kur (g2)              se               z               p
##       0.6420749       0.6859943       0.9359770       0.3492850

kurtosis.slight
```

```
## Excess Kur (g2)              se              z              p
##    4.339628e+00    6.859943e-01    6.326041e+00    2.515306e-10
```

The above analysis shows that the z-value of skewness (skew/SE) is between -7 and +7. Since the number of observations are greater than 300 (n>300) and z-values lay between -7 and +7, the above distribution can be considered normal Negative skewness of -1.75 implies the histogram is left skewed Kurtosis greater than 3 (5.94) implies lepokurtic condition

```
#------Question 1: Variation of accident severity across police forces-------
--

#--------------------------------------Chi-squared test-----------------------
---

chisq.test(x=Accident.cleaned$Police_Force,y=Accident.cleaned$Accident_Severi
ty)
```

```
## Warning in chisq.test(x = Accident.cleaned$Police_Force, y =
## Accident.cleaned$Accident_Severity): Chi-squared approximation may be
incorrect
```

```
##
##  Pearson's Chi-squared test
##
## data:  Accident.cleaned$Police_Force and
Accident.cleaned$Accident_Severity
## X-squared = 1874.9, df = 100, p-value < 2.2e-16
```

```
#Analysing the residual data where std. res>1.96 and std. res <-1.96

CrossTable(x=Accident.cleaned$Police_Force,y=Accident.cleaned$Accident_Severi
ty,expected = TRUE,        prop.c = FALSE,        prop.t = FALSE,
prop.chisq = FALSE,        chisq = TRUE,        sresid = TRUE)
```

```
## Warning in chisq.test(tab, correct = FALSE, ...): Chi-squared
approximation may
## be incorrect
```

```
##    Cell Contents
## |-------------------------|
## |                       N |
## |              Expected N |
## |           N / Row Total |
## |            Std Residual |
## |-------------------------|
##
## ======================================================================
##                                   Accident.cleaned$Accident_Severity
## Accident.cleaned$Police_Force      Fatal    Serious     Slight     Total
## ----------------------------------------------------------------------
```

```
## Metropolitan Police                    129       3566      23052     26747
##                                      344.9     4636.9    21765.2
##                                      0.005      0.133      0.862     0.206
##                                    -11.625    -15.727      8.722
## -----------------------------------------------------------------------
## Cumbria                                 27        250       1014      1291
##                                       16.6      223.8     1050.5
##                                      0.021      0.194      0.785     0.010
##                                      2.538      1.751     -1.127
## -----------------------------------------------------------------------
## Lancashire                              40        641       2674      3355
##                                       43.3      581.6     2730.1
##                                      0.012      0.191      0.797     0.026
##                                     -0.496      2.462     -1.074
## -----------------------------------------------------------------------
## Merseyside                              26        500       1715      2241
##                                       28.9      388.5     1823.6
##                                      0.012      0.223      0.765     0.017
##                                     -0.539      5.657     -2.543
## -----------------------------------------------------------------------
## Greater Manchester                      50        687       3332      4069
##                                       52.5      705.4     3311.1
##                                      0.012      0.169      0.819     0.031
##                                     -0.340     -0.693      0.363
## -----------------------------------------------------------------------
## Cheshire                                21        327       1862      2210
##                                       28.5      383.1     1798.4
##                                      0.010      0.148      0.843     0.017
##                                     -1.404     -2.868      1.500
## -----------------------------------------------------------------------
## Northumbria                             31        460       1871      2362
##                                       30.5      409.5     1922.1
##                                      0.013      0.195      0.792     0.018
##                                      0.099      2.496     -1.165
## -----------------------------------------------------------------------
## Durham                                  19        204        825      1048
##                                       13.5      181.7      852.8
##                                      0.018      0.195      0.787     0.008
##                                      1.493      1.656     -0.952
## -----------------------------------------------------------------------
## North Yorkshire                         38        361       1457      1856
##                                       23.9      321.8     1510.3
##                                      0.020      0.195      0.785     0.014
##                                      2.876      2.188     -1.372
## -----------------------------------------------------------------------
## West Yorkshire                          39        742       3589      4370
##                                       56.3      757.6     3556.1
##                                      0.009      0.170      0.821     0.034
##                                     -2.311     -0.567      0.552
## -----------------------------------------------------------------------
```

```
## South Yorkshire                        42         692        2060       2794
##                                        36.0       484.4      2273.6
##                                        0.015      0.248      0.737      0.021
##                                        0.995      9.434      -4.480
## -----------------------------------------------------------------------------
## Humberside                             28         475        1825       2328
##                                        30.0       403.6      1894.4
##                                        0.012      0.204      0.784      0.018
##                                        -0.368     3.555      -1.594
## -----------------------------------------------------------------------------
## Cleveland                              7          137        535        679
##                                        8.8        117.7      552.5
##                                        0.010      0.202      0.788      0.005
##                                        -0.593     1.778      -0.746
## -----------------------------------------------------------------------------
## West Midlands                          54         885        4743       5682
##                                        73.3       985.0      4623.7
##                                        0.010      0.156      0.835      0.044
##                                        -2.251     -3.188     1.755
## -----------------------------------------------------------------------------
## Staffordshire                          31         217        1559       1807
##                                        23.3       313.3      1470.4
##                                        0.017      0.120      0.863      0.014
##                                        1.595      -5.439     2.310
## -----------------------------------------------------------------------------
## West Mercia                            53         438        1630       2121
##                                        27.3       367.7      1725.9
##                                        0.025      0.207      0.769      0.016
##                                        4.905      3.666      -2.310
## -----------------------------------------------------------------------------
## Warwickshire                           34         271        1063       1368
##                                        17.6       237.2      1113.2
##                                        0.025      0.198      0.777      0.011
##                                        3.896      2.197      -1.505
## -----------------------------------------------------------------------------
## Derbyshire                             34         297        1312       1643
##                                        21.2       284.8      1337.0
##                                        0.021      0.181      0.799      0.013
##                                        2.784      0.721      -0.683
## -----------------------------------------------------------------------------
## Nottinghamshire                        32         394        2066       2492
##                                        32.1       432.0      2027.8
##                                        0.013      0.158      0.829      0.019
##                                        -0.023     -1.829     0.847
## -----------------------------------------------------------------------------
## Lincolnshire                           45         454        1409       1908
##                                        24.6       330.8      1552.6
##                                        0.024      0.238      0.738      0.015
##                                        4.112      6.775      -3.645
## -----------------------------------------------------------------------------
```

```
## Leicestershire                            19        248       1356      1623
##                                         20.9      281.4     1320.7
##                                        0.012      0.153      0.835     0.012
##                                       -0.421     -1.989      0.971
## ----------------------------------------------------------------------
## Northamptonshire                          40        207        821      1068
##                                         13.8      185.2      869.1
##                                        0.037      0.194      0.769     0.008
##                                        7.068      1.606     -1.631
## ----------------------------------------------------------------------
## Cambridgeshire                            42        377       1585      2004
##                                         25.8      347.4     1630.7
##                                        0.021      0.188      0.791     0.015
##                                        3.179      1.587     -1.133
## ----------------------------------------------------------------------
## Norfolk                                   28        351       1388      1767
##                                         22.8      306.3     1437.9
##                                        0.016      0.199      0.786     0.014
##                                        1.093      2.552     -1.316
## ----------------------------------------------------------------------
## Suffolk                                   31        233       1309      1573
##                                         20.3      272.7     1280.0
##                                        0.020      0.148      0.832     0.012
##                                        2.380     -2.404      0.810
## ----------------------------------------------------------------------
## Bedfordshire                              19        250       1390      1659
##                                         21.4      287.6     1350.0
##                                        0.011      0.151      0.838     0.013
##                                       -0.517     -2.218      1.089
## ----------------------------------------------------------------------
## Hertfordshire                             23        351       1868      2242
##                                         28.9      388.7     1824.4
##                                        0.010      0.157      0.833     0.017
##                                       -1.099     -1.911      1.020
## ----------------------------------------------------------------------
## Essex                                     45        806       2384      3235
##                                         41.7      560.8     2632.5
##                                        0.014      0.249      0.737     0.025
##                                        0.509     10.353     -4.843
## ----------------------------------------------------------------------
## Thames Valley                             59        677       3393      4129
##                                         53.2      715.8     3359.9
##                                        0.014      0.164      0.822     0.032
##                                        0.789     -1.451      0.570
## ----------------------------------------------------------------------
## Hampshire                                 42        902       3128      4072
##                                         52.5      705.9     3313.6
##                                        0.010      0.222      0.768     0.031
##                                       -1.450      7.379     -3.224
## ----------------------------------------------------------------------
```

```
## Surrey                                     33       482      2987      3502
##                                          45.2     607.1    2849.7
##                                         0.009     0.138     0.853     0.027
##                                        -1.809    -5.078     2.571
## -----------------------------------------------------------------------------
## Kent                                       56       758      3974      4788
##                                          61.7     830.1    3896.2
##                                         0.012     0.158     0.830     0.037
##                                        -0.730    -2.501     1.246
## -----------------------------------------------------------------------------
## Sussex                                     50       849      3013      3912
##                                          50.4     678.2    3183.4
##                                         0.013     0.217     0.770     0.030
##                                        -0.062     6.559    -3.020
## -----------------------------------------------------------------------------
## City of London                              2        56       255       313
##                                           4.0      54.3     254.7
##                                         0.006     0.179     0.815     0.002
##                                        -1.013     0.236     0.019
## -----------------------------------------------------------------------------
## Devon and Cornwall                         54       714      2800      3568
##                                          46.0     618.6    2903.4
##                                         0.015     0.200     0.785     0.027
##                                         1.179     3.838    -1.920
## -----------------------------------------------------------------------------
## Avon and Somerset                          46       295      2433      2774
##                                          35.8     480.9    2257.3
##                                         0.017     0.106     0.877     0.021
##                                         1.711    -8.477     3.698
## -----------------------------------------------------------------------------
## Gloucestershire                            19       255       525       799
##                                          10.3     138.5     650.2
##                                         0.024     0.319     0.657     0.006
##                                         2.710     9.897    -4.909
## -----------------------------------------------------------------------------
## Wiltshire                                  23       225      1072      1320
##                                          17.0     228.8    1074.1
##                                         0.017     0.170     0.812     0.010
##                                         1.449    -0.254    -0.065
## -----------------------------------------------------------------------------
## Dorset                                     26       291      1279      1596
##                                          20.6     276.7    1298.7
##                                         0.016     0.182     0.801     0.012
##                                         1.195     0.861    -0.548
## -----------------------------------------------------------------------------
## North Wales                                21       238       745      1004
##                                          12.9     174.1     817.0
##                                         0.021     0.237     0.742     0.008
##                                         2.239     4.847    -2.519
## -----------------------------------------------------------------------------
```

```
## Gwent                                     19        94       516       629
##                                          8.1     109.0     511.8
##                                        0.030     0.149     0.820     0.005
##                                        3.824    -1.441     0.184
## -----------------------------------------------------------------------------
## South Wales                               32       239      1460      1731
##                                         22.3     300.1    1408.6
##                                        0.018     0.138     0.843     0.013
##                                        2.049    -3.527     1.370
## -----------------------------------------------------------------------------
## Dyfed-Powys                               26       265       899      1190
##                                         15.3     206.3     968.4
##                                        0.022     0.223     0.755     0.009
##                                        2.720     4.087    -2.229
## -----------------------------------------------------------------------------
## Northern                                  17        63       272       352
##                                          4.5      61.0     286.4
##                                        0.048     0.179     0.773     0.003
##                                        5.849     0.253    -0.853
## -----------------------------------------------------------------------------
## Grampian                                  14       149       304       467
##                                          6.0      81.0     380.0
##                                        0.030     0.319     0.651     0.004
##                                        3.251     7.562    -3.900
## -----------------------------------------------------------------------------
## Tayside                                   22       120       317       459
##                                          5.9      79.6     373.5
##                                        0.048     0.261     0.691     0.004
##                                        6.610     4.532    -2.924
## -----------------------------------------------------------------------------
## Fife                                       5        72       239       316
##                                          4.1      54.8     257.1
##                                        0.016     0.228     0.756     0.002
##                                        0.458     2.326    -1.131
## -----------------------------------------------------------------------------
## Lothian and Borders                       22       294      1376      1692
##                                         21.8     293.3    1376.9
##                                        0.013     0.174     0.813     0.013
##                                        0.039     0.039    -0.023
## -----------------------------------------------------------------------------
## Central                                    6        88       312       406
##                                          5.2      70.4     330.4
##                                        0.015     0.217     0.768     0.003
##                                        0.334     2.100    -1.011
## -----------------------------------------------------------------------------
## Strathclyde                               44       544      2597      3185
##                                         41.1     552.2    2591.8
##                                        0.014     0.171     0.815     0.025
##                                        0.458    -0.347     0.103
## -----------------------------------------------------------------------------
```

```
## Dumfries and Galloway                     11         43        182       236
##                                          3.0       40.9      192.0
##                                        0.047      0.182      0.771     0.002
##                                        4.561      0.326     -0.725
## -----------------------------------------------------------------------
## Total                                   1676      22534     105772    129982
## =======================================================================
##
## Statistics for All Table Factors
##
## Pearson's Chi-squared test
## ------------------------------------------------------------
## Chi^2 = 1874.853      d.f. = 100      p <2e-16
```

```
#-------------------------------------ANOVA test-----------------------------
---

Accident$Police_Force=as.factor(Accident$Police_Force)
Accident$Accident_Severity=as.factor(Accident$Accident_Severity)
acc.old.anova<-Accident%>%
  drop_na(Accident_Severity)%>%
  group_by(Police_Force,Accident_Severity)%>%
  summarise(num.acc=n())
```

```
## `summarise()` has grouped output by 'Police_Force'. You can override using
the `.groups` argument.
```

```
two.way.acc<-aov(num.acc ~ Police_Force+Accident_Severity,data =
acc.old.anova)
summary(two.way.acc)
```

```
##                    Df    Sum Sq  Mean Sq F value   Pr(>F)
## Police_Force       50 227261752  4545235   1.559   0.0306 *
## Accident_Severity   2 118951621 59475811  20.399 3.72e-08 ***
## Residuals         100 291561061  2915611
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
pol.p<-1-pf(1.559,50,100)
sev.p<-1-pf(20.399,2,100)
pol.p
```

```
## [1] 0.03059033
```

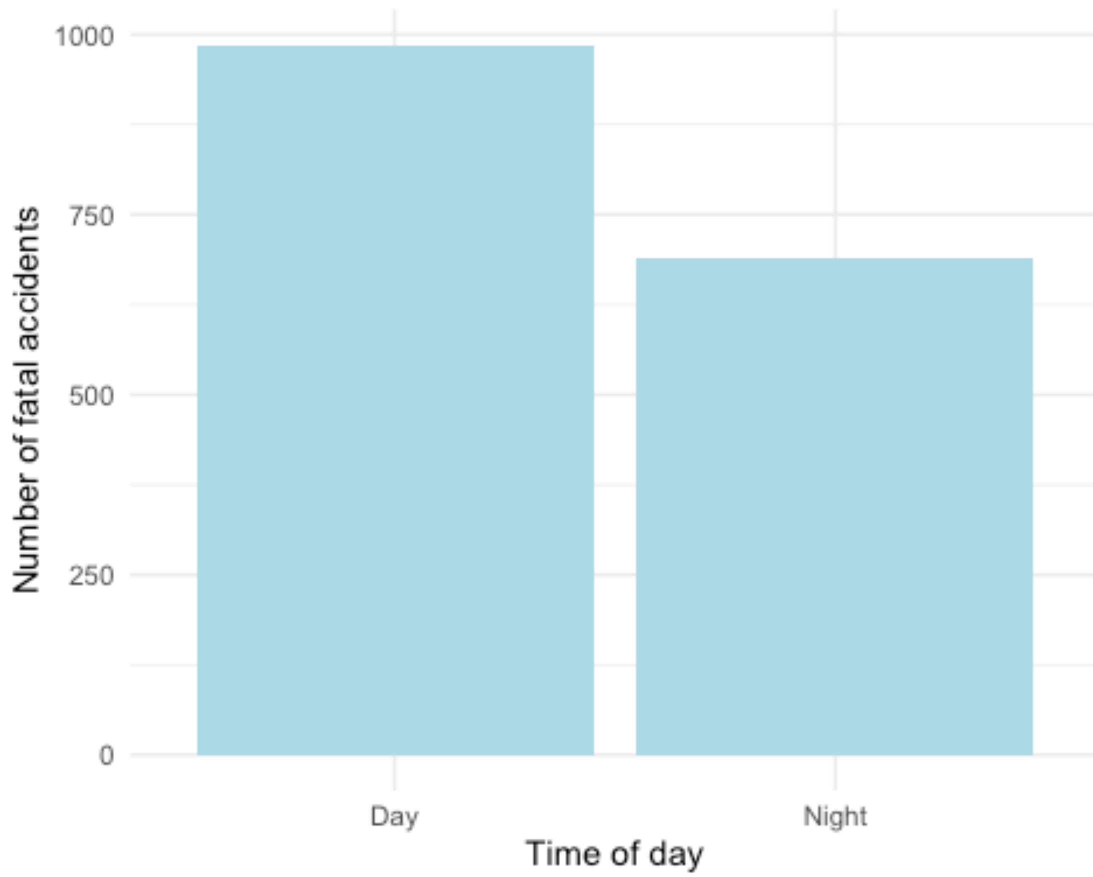
```
sev.p
```

```
## [1] 3.716856e-08
```

```
#Question 2: Testing significance of accidents occuring on weekdays and
weekends

# Let p1= proportion of accidents on weekdays (0.7522)
```

```r
# Let p2= proportion of accidents on weekends (0.2477)
# H0: p2-p1=0
# Ha:p2-p1!=0
# z-value is large (297.92) implies p-value tends to zero. Hence, null
hypothesis can be rejected
# The above test was by hand calculations
# --------------------Perform similar test for casualties in r---------------
-
# Let x1= Total casualties on weekdays
# Let x2= Total casualties on weekends
# H0: x1-x2=0
# Ha: x1-x2!=0
#--------Distinguishing weekday and weekend-----------------
Accident.cleaned<-Accident.cleaned%>%

mutate(Day_Type=if_else(Day_of_Week=="Monday"|Day_of_Week=="Tuesday"|Day_of_W
eek=="Wednesday"|Day_of_Week=="Thursday"|Day_of_Week=="Friday","Weekday","Wee
kend"))
Accident.cleaned%>%
  group_by(Day_Type)%>%
  summarise(prob=100*n()/nrow(Accident.cleaned))

## # A tibble: 2 × 2
##   Day_Type  prob
##   <chr>    <dbl>
## 1 Weekday   75.2
## 2 Weekend   24.8

table(Accident.cleaned$Day_Type)

##
## Weekday Weekend
##   97779   32203

#----------T-test----------------------------------------
t.test(formula=Accident.cleaned$Number_of_Casualties~Accident.cleaned$Day_Typ
e)

##
##  Welch Two Sample t-test
##
## data:  Accident.cleaned$Number_of_Casualties by Accident.cleaned$Day_Type
## t = -18.896, df = 48696, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group Weekday and
group Weekend is not equal to 0
## 95 percent confidence interval:
##  -0.11069482 -0.08988854
## sample estimates:
## mean in group Weekday mean in group Weekend
##              1.290666              1.390957
```

```
t.test(formula=Accident.cleaned$Number_of_Vehicles~Accident.cleaned$Day_Type)

##
##  Welch Two Sample t-test
##
## data:  Accident.cleaned$Number_of_Vehicles by Accident.cleaned$Day_Type
## t = 8.0514, df = 53493, p-value = 8.351e-16
## alternative hypothesis: true difference in means between group Weekday and
group Weekend is not equal to 0
## 95 percent confidence interval:
##  0.02872765 0.04721479
## sample estimates:
## mean in group Weekday mean in group Weekend
##              1.847554              1.809583

#Question 3: Investigate whether more fatal accidents occur in the day or
night



#----------Distinguishing day and night--------------

Accident.cleaned<-Accident.cleaned%>%
  mutate(condition=substr(Accident.cleaned$Light_Conditions,1,3))%>%
  mutate(day.dark=if_else(condition=="Dar","Night","Day"))

#------Graphical and statistical analysis-------------

Accident.cleaned%>%
  filter(Accident_Severity=="Fatal")%>%

ggplot(aes(x=day.dark))+geom_bar(fill="lightblue")+theme_minimal()+labs(x="Ti
me of day",y="Number of fatal accidents")
```




```
Accident.cleaned%>%
  filter(Accident_Severity=="Fatal")%>%
  drop_na(Light_Conditions)%>%
  group_by(Accident_Severity,Light_Conditions)%>%
  summarise(Number_of_accidents=n(),
            Total_casualties=sum(Number_of_Casualties),
            casualty_per_acc=mean(Number_of_Casualties),
            veh.total=sum(Number_of_Vehicles),
            veh.stat=mean(Number_of_Vehicles))
```

```
## `summarise()` has grouped output by 'Accident_Severity'. You can override
using the `.groups` argument.
```

```
## # A tibble: 5 × 7
## # Groups:   Accident_Severity [1]
##   Accident_Severity Light_Conditions         Number_of_acciden…
Total_casualties
##   <fct>             <fct>                                 <int>
<int>
## 1 Fatal             Daylight                                986
1625
## 2 Fatal             Darkness - lights lit                   328
533
```

```
## 3 Fatal             Darkness - lights unlit                    21
32
## 4 Fatal             Darkness - no lighting                    317
600
## 5 Fatal             Darkness - lighting unk…                   24
33
## # … with 3 more variables: casualty_per_acc <dbl>, veh.total <int>,
## #   veh.stat <dbl>
```

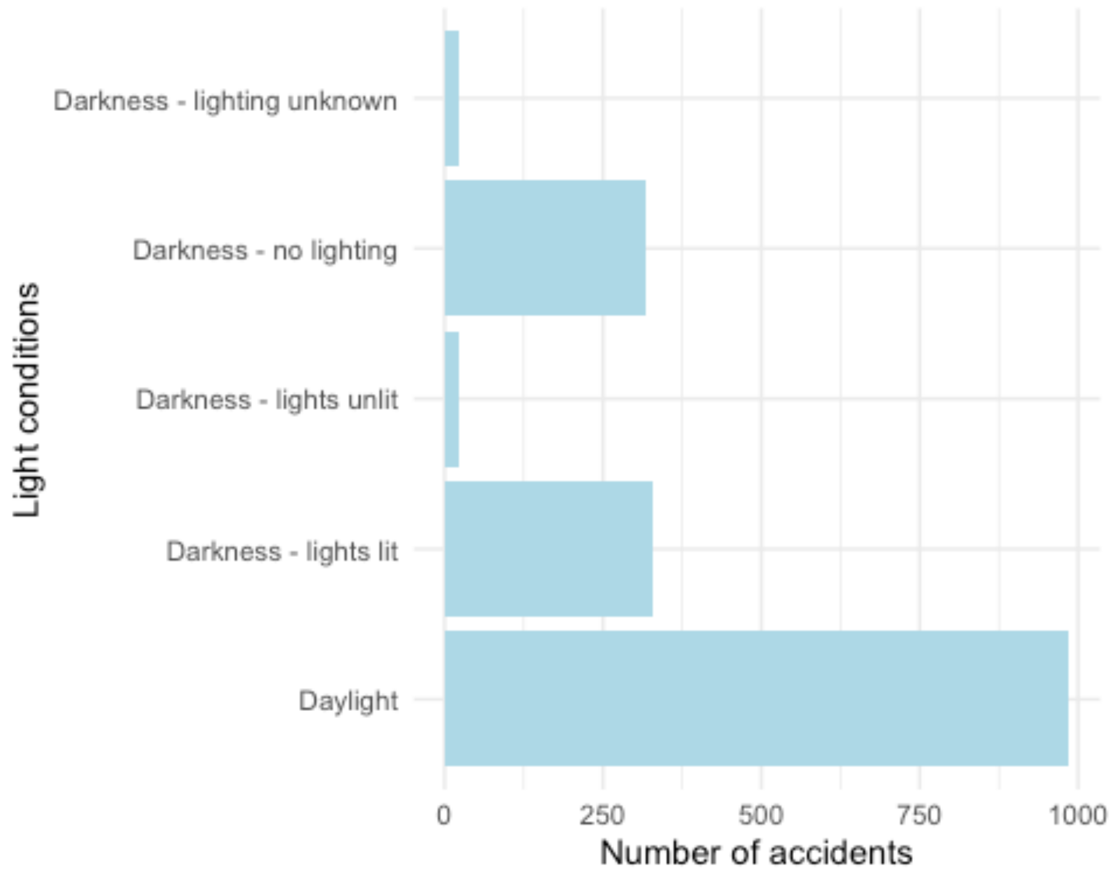
```
Accident.cleaned%>%
  filter(Accident_Severity=="Fatal")%>%
  drop_na(day.dark)%>%
  group_by(Accident_Severity,day.dark,Speed_limit)%>%
  summarise(Number_of_accidents=n(),
            Total_casualties=sum(Number_of_Casualties),
            casualty_per_acc=mean(Number_of_Casualties),
            veh.total=sum(Number_of_Vehicles),
            veh.stat=mean(Number_of_Vehicles))
```

```
## `summarise()` has grouped output by 'Accident_Severity', 'day.dark'. You
can override using the `.groups` argument.
```

```
## # A tibble: 12 × 8
## # Groups:   Accident_Severity, day.dark [2]
##    Accident_Severity day.dark Speed_limit Number_of_accidents
Total_casualties
##    <fct>             <chr>          <int>               <int>
<int>
##  1 Fatal             Day               20                  34
38
##  2 Fatal             Day               30                 317
398
##  3 Fatal             Day               40                  87
136
##  4 Fatal             Day               50                  92
189
##  5 Fatal             Day               60                 373
708
##  6 Fatal             Day               70                  83
156
##  7 Fatal             Night             20                  18
26
##  8 Fatal             Night             30                 248
372
##  9 Fatal             Night             40                  61
102
## 10 Fatal             Night             50                  52
90
## 11 Fatal             Night             60                 210
413
```

```
## 12 Fatal              Night              70                  101
195
## # … with 3 more variables: casualty_per_acc <dbl>, veh.total <int>,
## #   veh.stat <dbl>
```

```
Accident.cleaned%>%
  drop_na(Light_Conditions)%>%
  filter(Accident_Severity=="Fatal")%>%

ggplot(aes(x=Light_Conditions))+geom_bar(fill="lightblue")+theme_minimal()+la
bs(x="Light conditions",y="Number of accidents")+coord_flip()
```




```
Accident.cleaned%>%
  drop_na(Light_Conditions)%>%
  group_by(Accident_Severity,Light_Conditions)%>%
  summarise(n(),
            Total_casualties=sum(Number_of_Casualties),
            casualty_per_acc=mean(Number_of_Casualties))
```

```
## `summarise()` has grouped output by 'Accident_Severity'. You can override
using the `.groups` argument.
```

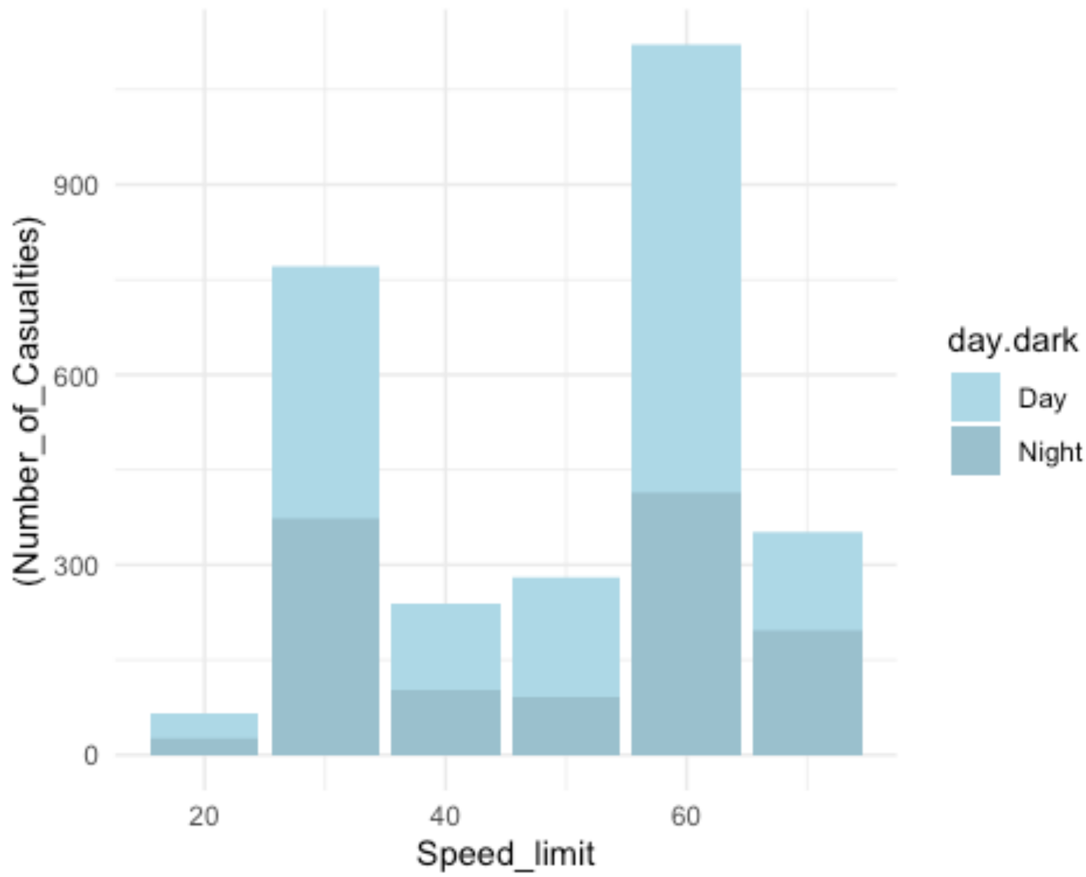
```
## # A tibble: 15 × 5
## # Groups:   Accident_Severity [3]
```

```
##    Accident_Severity Light_Conditions            `n()` Total_casualties
casualty_per_acc
##    <fct>             <fct>                       <int>            <int>
<dbl>
##  1 Fatal             Daylight                      986             1625
1.65
##  2 Fatal             Darkness - lights lit         328              533
1.62
##  3 Fatal             Darkness - lights unlit        21               32
1.52
##  4 Fatal             Darkness - no lighting        317              600
1.89
##  5 Fatal             Darkness - lighting unknown    24               33
1.38
##  6 Serious           Daylight                    15463            21055
1.36
##  7 Serious           Darkness - lights lit        4896             6541
1.34
##  8 Serious           Darkness - lights unlit       161              241
1.50
##  9 Serious           Darkness - no lighting       1569             2718
1.73
## 10 Serious           Darkness - lighting unknown   445              602
1.35
## 11 Slight            Daylight                    76229            98245
1.29
## 12 Slight            Darkness - lights lit       21623            28027
1.30
## 13 Slight            Darkness - lights unlit       747              987
1.32
## 14 Slight            Darkness - no lighting       4792             6805
1.42
## 15 Slight            Darkness - lighting unknown  2380             2948
1.24
```

```
Accident.cleaned%>%
  filter(Light_Conditions=="Darkness - lights
lit"|Light_Conditions=="Darkness - lights unlit"|Light_Conditions=="Darkness
- no lighting"|Light_Conditions=="Darkness - lighting unknown")%>%
  summarise(n())
```

```
##     n()
## 1 37303
```

```
Accident.cleaned%>%
  filter(Accident_Severity=="Fatal")%>%

ggplot(aes(x=Speed_limit,y=(Number_of_Casualties),fill=day.dark))+geom_col()+
theme_minimal()+scale_fill_manual(values = c("lightblue","lightblue3"))
```



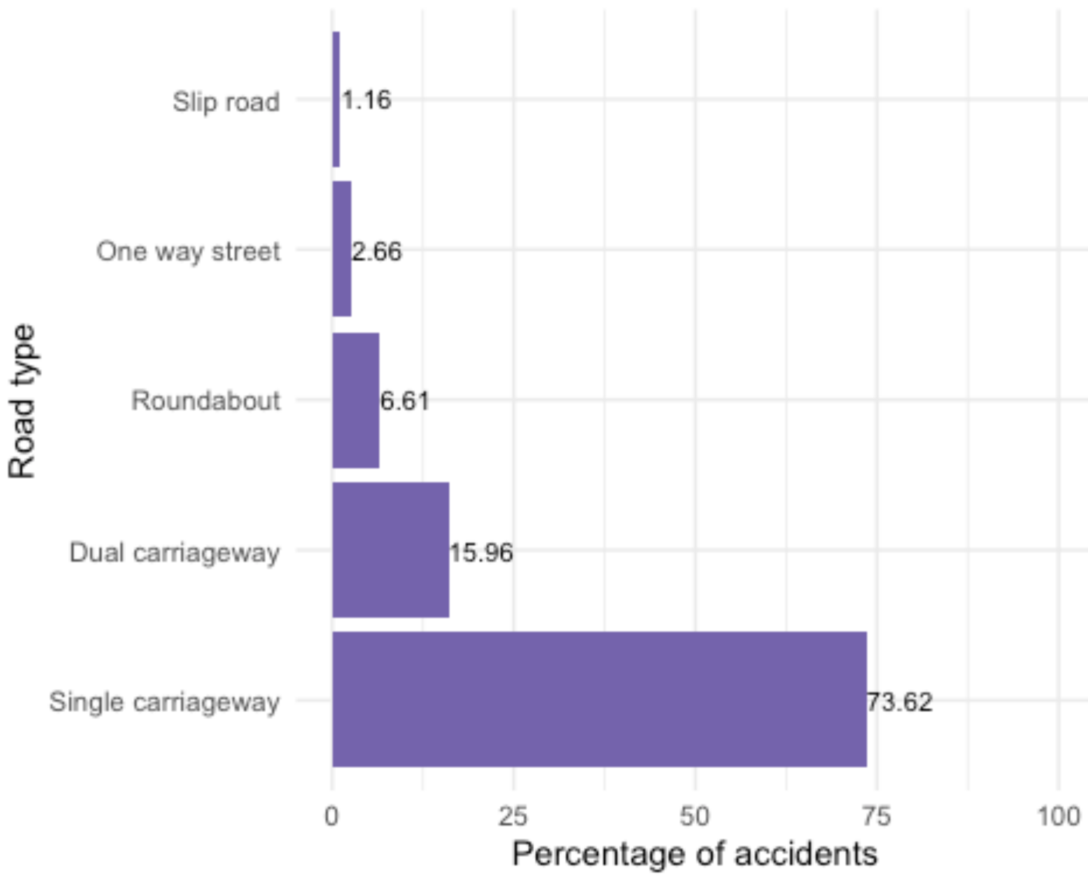

```
#---------T-test for significance------------

Accident.cleaned%>%
  drop_na(day.dark)%>%
  filter(Accident_Severity=="Fatal")%>%
  group_by(day.dark)%>%
  summarise(acc.tot=n(),
            cas.tot=sum(Number_of_Casualties))

## # A tibble: 2 × 3
##   day.dark acc.tot cas.tot
##   <chr>      <int>   <int>
## 1 Day          986    1625
## 2 Night        690    1198

t.test(formula=Accident.cleaned$Number_of_Casualties~Accident.cleaned$day.dar
k)

##
##  Welch Two Sample t-test
##
## data:  Accident.cleaned$Number_of_Casualties by Accident.cleaned$day.dark
## t = -7.8674, df = 67093, p-value = 3.674e-15
## alternative hypothesis: true difference in means between group Day and
```

```
group Night is not equal to 0
## 95 percent confidence interval:
##  -0.04669795 -0.02807086
## sample estimates:
##   mean in group Day mean in group Night
##            1.304786            1.342171
```

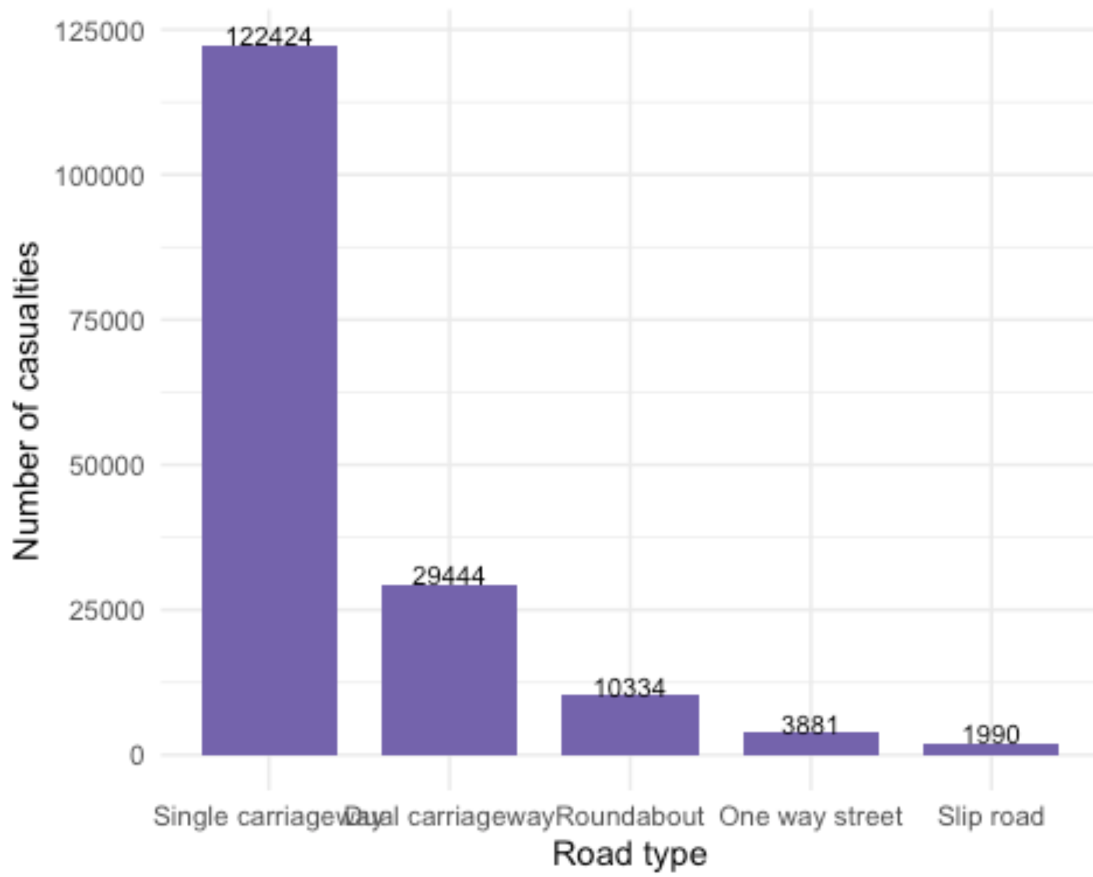
```
#Question 4: Investigating dangerous road types}



#---------------------Uni-variate analysis of Road_type----------------------

road.type.plot.data<-Accident.cleaned%>%
  drop_na(Road_Type)%>%
  group_by(Road_Type)%>%
  summarise(No_of_acc=n(),
            cas.total=sum(Number_of_Casualties),
            cas.stat=mean(Number_of_Casualties),
            veh.total=sum(Number_of_Vehicles),
            veh.stat=mean(Number_of_Vehicles))
road.type.plot.data
```

```
## # A tibble: 5 × 6
##   Road_Type          No_of_acc cas.total cas.stat veh.total veh.stat
##   <fct>                  <int>     <int>    <dbl>     <int>    <dbl>
## 1 Roundabout              8417     10334     1.23     15983     1.90
## 2 One way street          3386      3881     1.15      5484     1.62
## 3 Dual carriageway       20340     29444     1.45     41209     2.03
## 4 Single carriageway     93811    122424     1.31    169339     1.81
## 5 Slip road               1476      1990     1.35      2742     1.86
```

```
  road.type.plot.data%>%
    mutate(perc=100*No_of_acc/sum(No_of_acc))%>%
    ggplot(aes(x=reorder(x=Road_Type,X=-
perc),y=perc))+geom_col(fill="#7463AC")+theme_minimal()+labs(x="Road
type",y="Percentage of
accidents")+coord_flip()+geom_text(aes(label=sprintf("%0.2f", round(perc,
digits = 2)),hjust="left"),size = 3)+ylim(0,100)
```



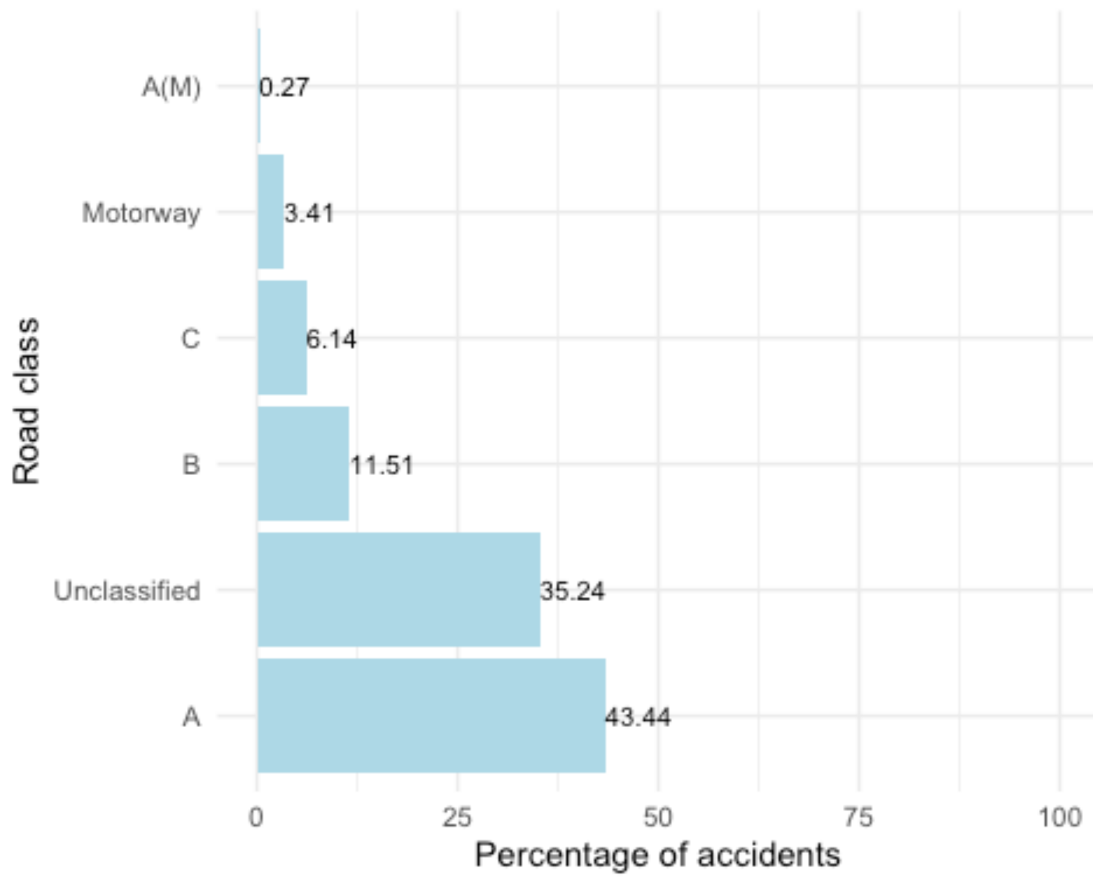

```
  road.type.plot.data%>%
    ggplot(aes(x=reorder(x=Road_Type,X=-
cas.total),y=cas.total))+geom_col(fill="#7463AC",width=0.75)+theme_minimal()+
labs(x="Road type",y="Number of
casualties")+geom_text(aes(label=cas.total,vjust="bottom"),size = 3)
```



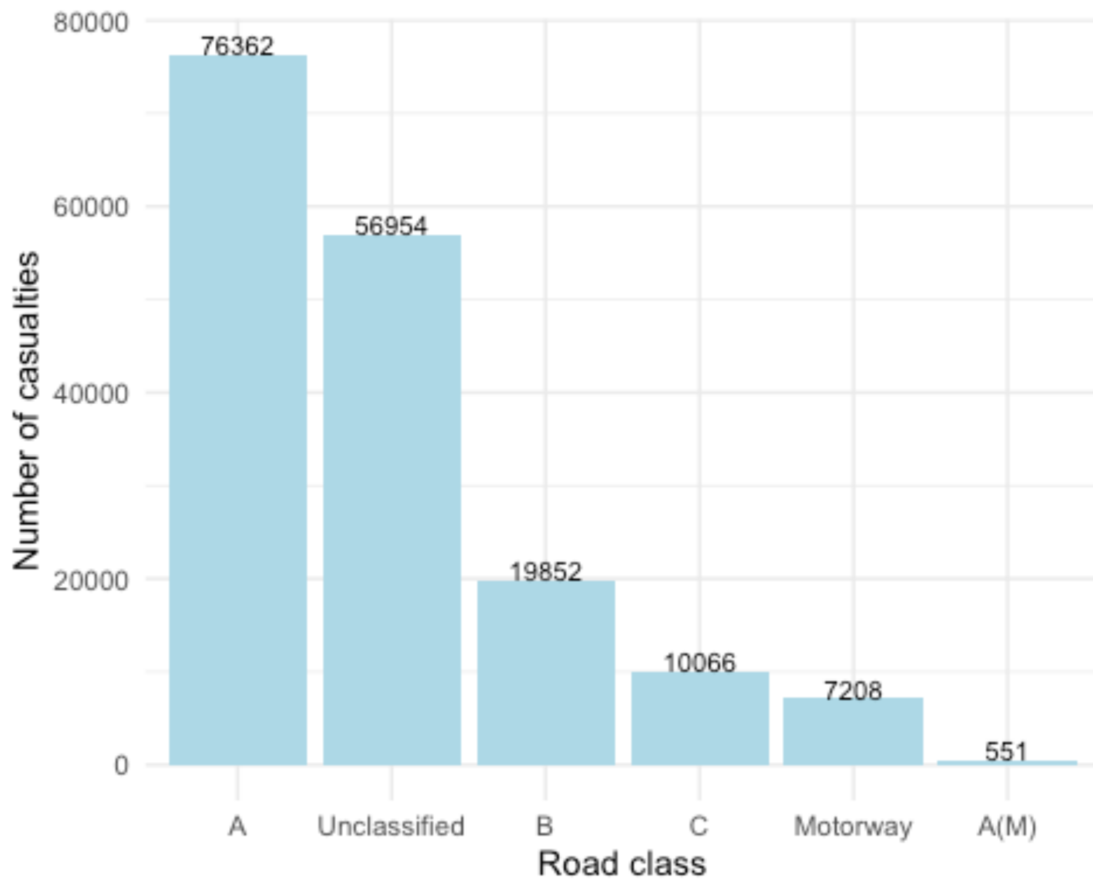

```
  oneway.test(formula =
Number_of_Casualties~Road_Type,data=Accident.cleaned,var.equal = TRUE)

##
##  One-way analysis of means
##
## data:  Number_of_Casualties and Road_Type
## F = 224.02, num df = 4, denom df = 127425, p-value < 2.2e-16

#------------------------Uni-variate analysis of Road class-----------------
-

road.class.plot.data<-Accident.cleaned%>%
  group_by(Road_Class_1st)%>%
  summarise(No_of_acc=n(),
            cas.total=sum(Number_of_Casualties),
            cas.stat=mean(Number_of_Casualties),
            veh.total=sum(Number_of_Vehicles),
            veh.stat=mean(Number_of_Vehicles))
road.class.plot.data

## # A tibble: 6 × 6
##   Road_Class_1st No_of_acc cas.total cas.stat veh.total veh.stat
##   <fct>              <int>     <int>    <dbl>     <int>    <dbl>
```

```
## 1 Motorway             4430      7208      1.63      10136      2.29
## 2 A(M)                  348       551      1.58        748      2.15
## 3 A                   56461     76362      1.35     107085      1.90
## 4 B                   14961     19852      1.33      27040      1.81
## 5 C                    7981     10066      1.26      13897      1.74
## 6 Unclassified        45801     56954      1.24      80020      1.75
```

```
road.class.plot.data%>%
  mutate(perc=100*No_of_acc/sum(No_of_acc))%>%
    ggplot(aes(x=reorder(x=Road_Class_1st,X=-
perc),y=perc))+geom_col(fill="lightblue")+theme_minimal()+labs(x="Road
class",y="Percentage of
accidents")+coord_flip()+geom_text(aes(label=sprintf("%0.2f", round(perc,
digits = 2)),hjust="left"),size = 3)+ylim(0,100)
```



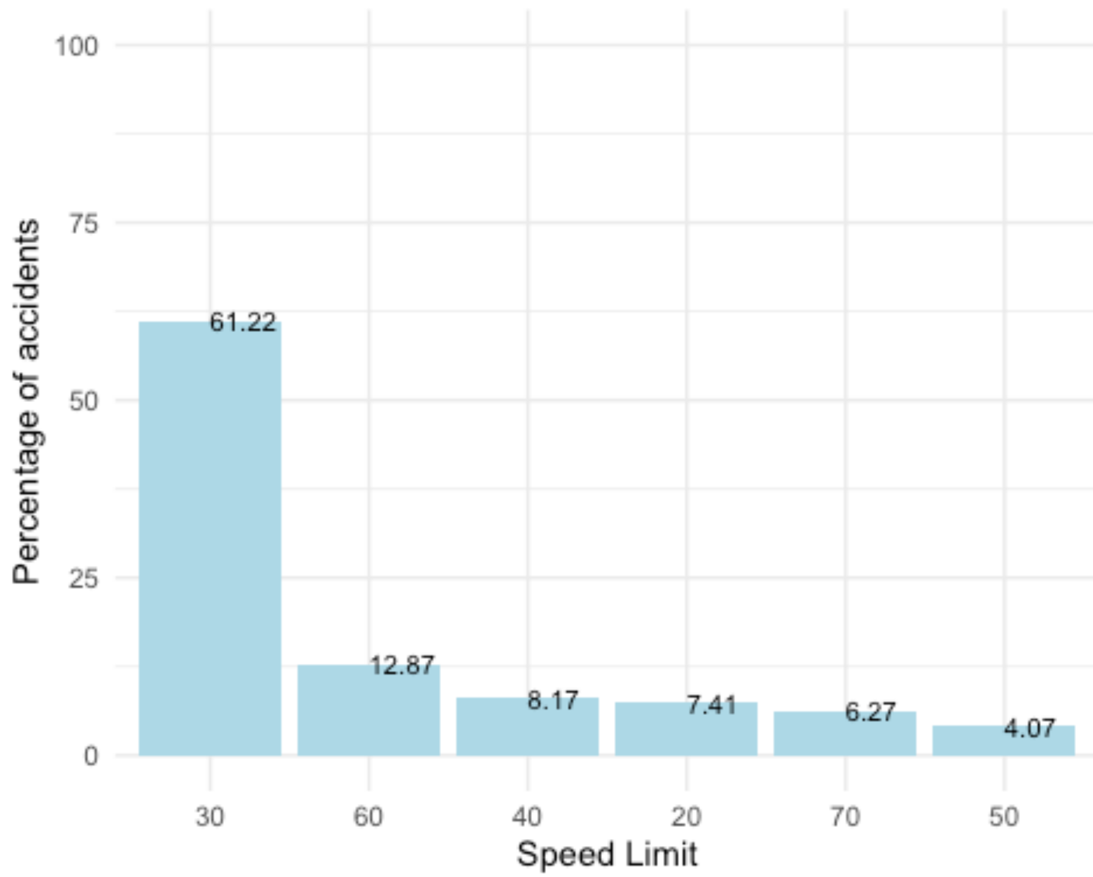

```
road.class.plot.data%>%
  ggplot(aes(x=reorder(x=Road_Class_1st,X=-
cas.total),y=cas.total))+geom_col(fill="lightblue")+theme_minimal()+labs(x="R
oad class",y="Number of
casualties")+geom_text(aes(label=cas.total,vjust="bottom"),size = 3)
```



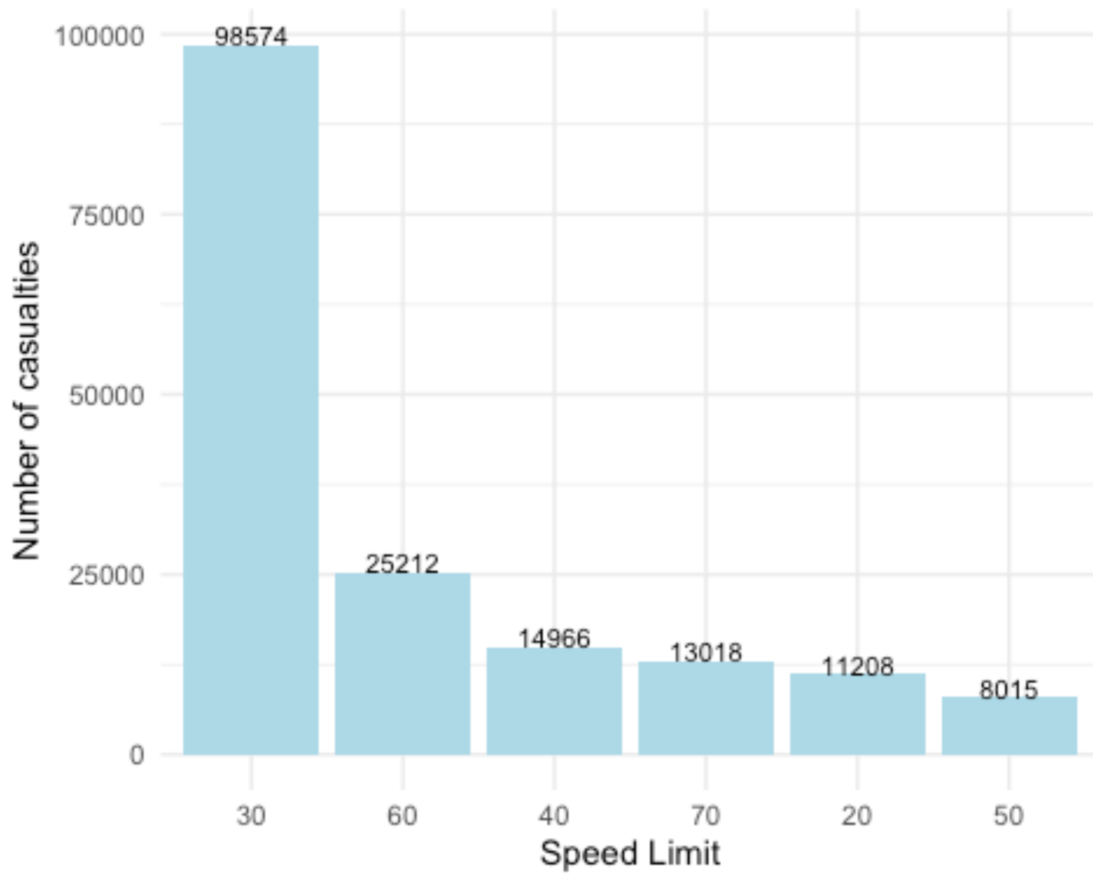

```
oneway.test(formula =
Number_of_Casualties~Road_Class_1st,data=Accident.cleaned,var.equal = TRUE)

##
##  One-way analysis of means
##
## data:  Number_of_Casualties and Road_Class_1st
## F = 274.23, num df = 5, denom df = 129976, p-value < 2.2e-16

#-----------------------------Uni-variate analysis of Speed limit-------------
--

speed.cas<-Accident.cleaned%>%
  group_by(Speed_limit)%>%
  summarise(No_of_acc=n(),
            cas.total=sum(Number_of_Casualties),
            cas.stat=mean(Number_of_Casualties),
            veh.total=sum(Number_of_Vehicles),
            veh.stat=mean(Number_of_Vehicles))
speed.cas

## # A tibble: 6 × 6
##   Speed_limit No_of_acc cas.total cas.stat veh.total veh.stat
##         <int>     <int>     <int>    <dbl>     <int>    <dbl>
```

```
## 1          20      9633     11208      1.16      16334      1.70
## 2          30     79569     98574      1.24     142710      1.79
## 3          40     10615     14966      1.41      20725      1.95
## 4          50      5286      8015      1.52      10927      2.07
## 5          60     16723     25212      1.51      30507      1.82
## 6          70      8156     13018      1.60      17723      2.17
```

```
speed.cas%>%
  mutate(perc=100*No_of_acc/sum(No_of_acc))%>%
    ggplot(aes(x=reorder(x=Speed_limit,X=-
perc),y=perc))+geom_col(fill="lightblue")+theme_minimal()+labs(x="Speed
Limit",y="Percentage of accidents")+geom_text(aes(label=sprintf("%0.2f",
round(perc, digits = 2)),hjust="left"),size = 3)+ylim(0,100)
```

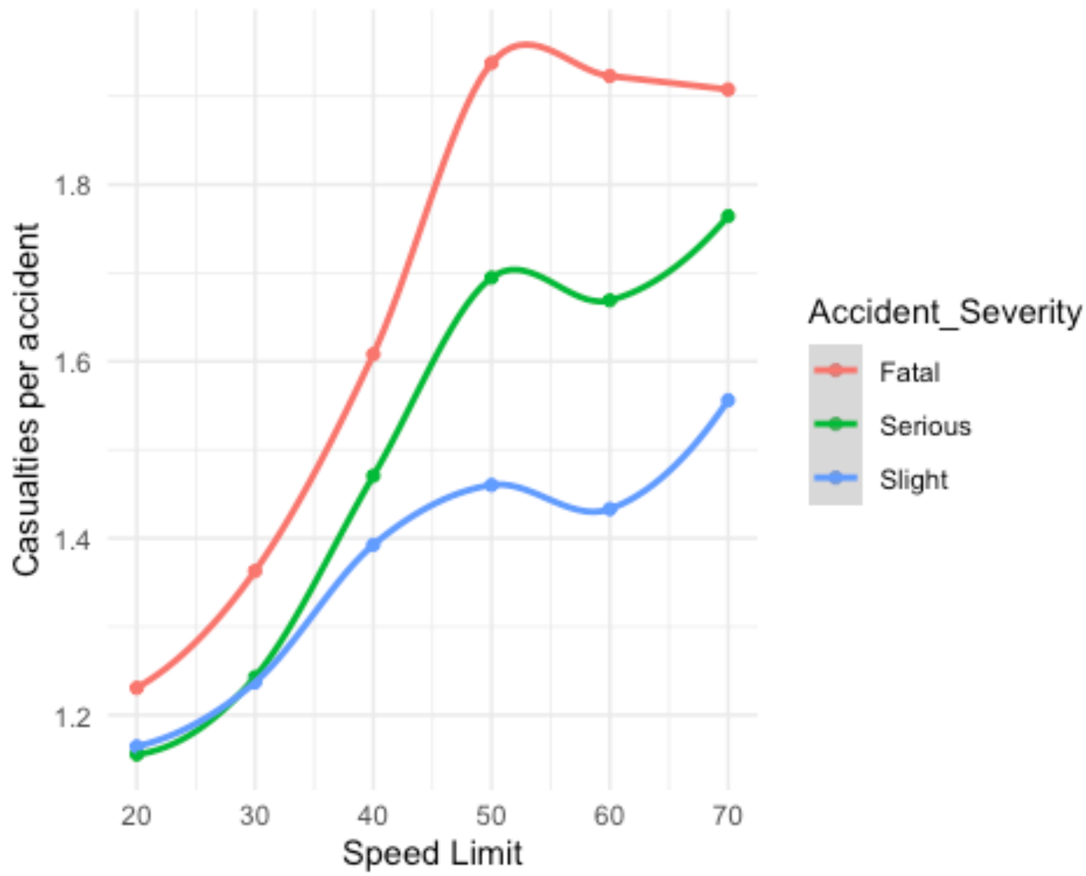



```
speed.cas%>%
  ggplot(aes(x=reorder(x=Speed_limit,X=-
cas.total),y=cas.total))+geom_col(fill="lightblue")+theme_minimal()+labs(x="S
peed Limit",y="Number of
casualties")+geom_text(aes(label=cas.total,vjust="bottom"),size = 3)
```



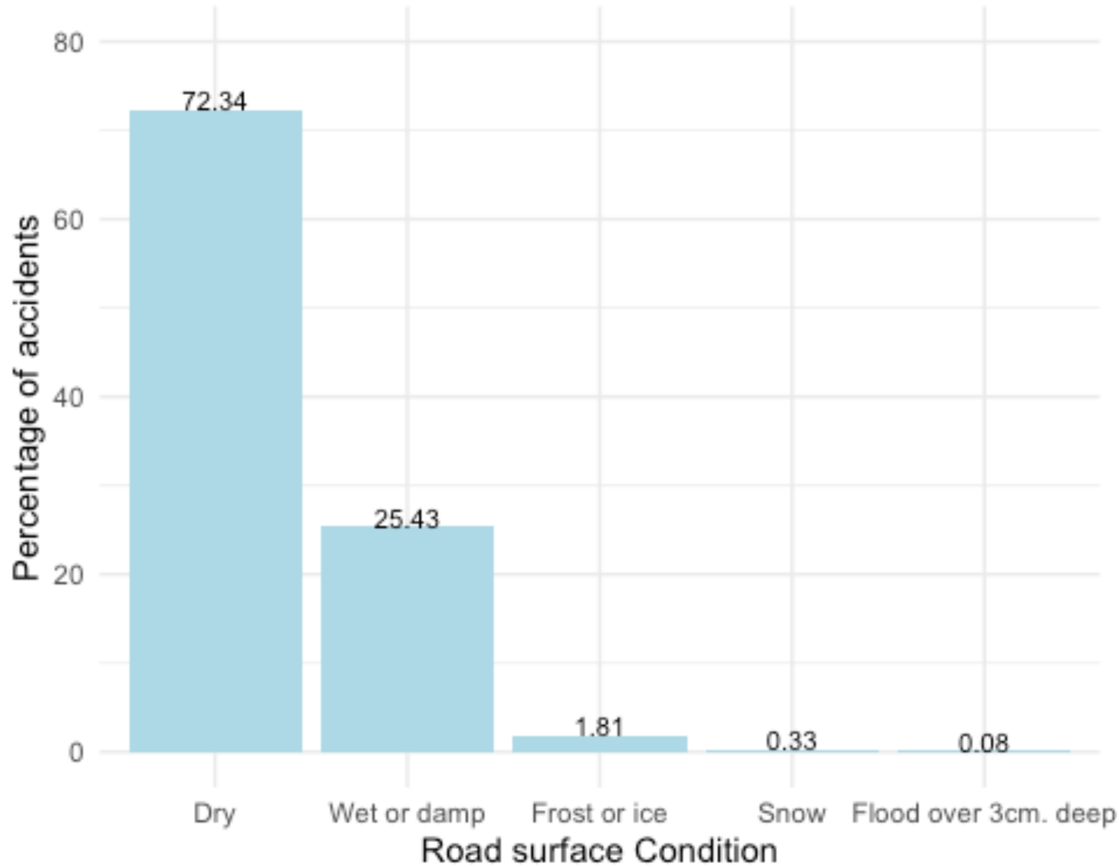

```
oneway.test(formula =
Number_of_Casualties~Speed_limit,data=Accident.cleaned,var.equal = TRUE)

##
##  One-way analysis of means
##
## data:  Number_of_Casualties and Speed_limit
## F = 793.97, num df = 5, denom df = 129976, p-value < 2.2e-16

speed.sev.cas<-Accident.cleaned%>%
  group_by(Speed_limit,Accident_Severity)%>%
  summarise(No_of_acc=n(),
            cas.total=sum(Number_of_Casualties),
            cas.stat=mean(Number_of_Casualties),
            veh.total=sum(Number_of_Vehicles),
            veh.stat=mean(Number_of_Vehicles))

## `summarise()` has grouped output by 'Speed_limit'. You can override using
the `.groups` argument.

speed.sev.cas%>%

ggplot(aes(x=Speed_limit,y=cas.stat,color=Accident_Severity))+geom_point()+th
eme_minimal()+stat_smooth()+labs(x="Speed Limit",y="Casualties per accident")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
## parametric, : Chernobyl! trL>n 6

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
## parametric, : Chernobyl! trL>n 6

## Warning in sqrt(sum.squares/one.delta): NaNs produced

## Warning in stats::qt(level/2 + 0.5, pred$df): NaNs produced

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
## parametric, : Chernobyl! trL>n 6

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
## parametric, : Chernobyl! trL>n 6

## Warning in sqrt(sum.squares/one.delta): NaNs produced

## Warning in stats::qt(level/2 + 0.5, pred$df): NaNs produced

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
## parametric, : Chernobyl! trL>n 6

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
## parametric, : Chernobyl! trL>n 6

## Warning in sqrt(sum.squares/one.delta): NaNs produced

## Warning in stats::qt(level/2 + 0.5, pred$df): NaNs produced

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max;
returning
## -Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max;
returning
## -Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max;
returning
## -Inf
```

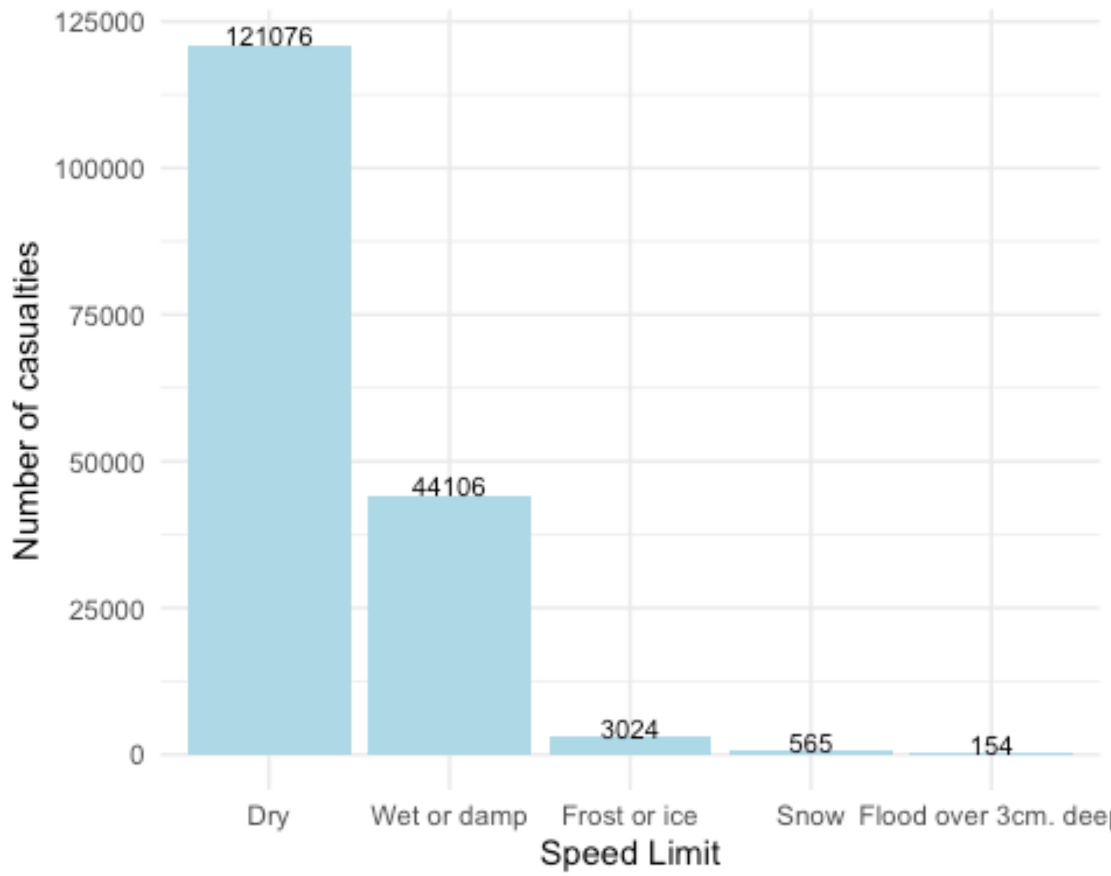



```
#-------------------Uni-variate analysis of Road surface--------------------

road.surf.plot.data<-Accident.cleaned%>%
  drop_na(Road_Surface_Conditions)%>%
  group_by(Road_Surface_Conditions)%>%
  summarise(No_of_acc=n(),
            cas.total=sum(Number_of_Casualties),
            cas.stat=mean(Number_of_Casualties),
            veh.total=sum(Number_of_Vehicles),
            veh.stat=mean(Number_of_Vehicles))
road.surf.plot.data

## # A tibble: 5 × 6
##   Road_Surface_Conditions No_of_acc cas.total cas.stat veh.total veh.stat
##   <fct>                       <int>     <int>    <dbl>     <int>    <dbl>
## 1 Dry                         92628    121076     1.31    171991     1.86
## 2 Wet or damp                 32564     44106     1.35     59148     1.82
## 3 Snow                          428       565     1.32       732     1.71
## 4 Frost or ice                 2319      3024     1.30      3711     1.60
## 5 Flood over 3cm. deep          106       154     1.45       161     1.52

road.surf.plot.data%>%
  mutate(perc=100*No_of_acc/sum(No_of_acc))%>%
```

```
    ggplot(aes(x=reorder(x=Road_Surface_Conditions,X=-
perc),y=perc))+geom_col(fill="lightblue")+theme_minimal()+labs(x="Road
surface Condition",y="Percentage of
accidents")+geom_text(aes(label=sprintf("%0.2f", round(perc, digits =
2)),vjust="bottom"),size = 3)+ylim(0,80)
```

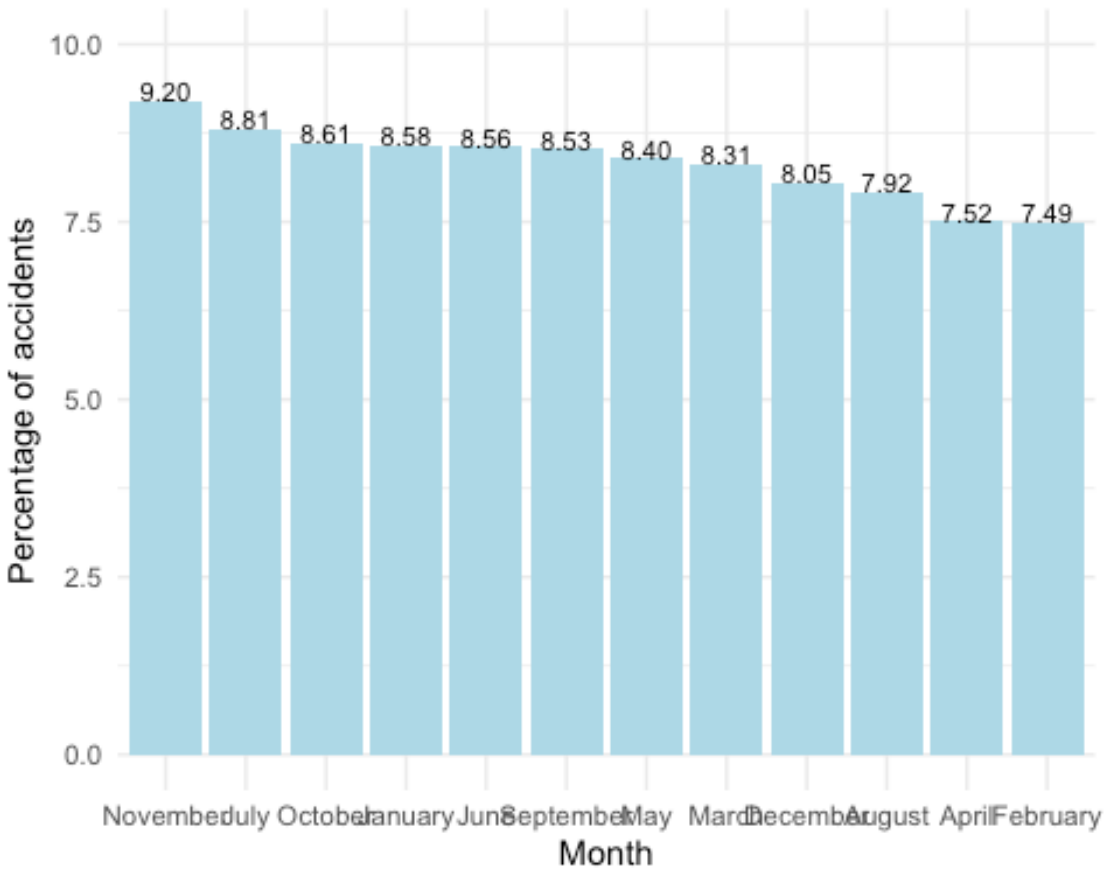



```
road.surf.plot.data%>%
  ggplot(aes(x=reorder(x=Road_Surface_Conditions,X=-
cas.total),y=cas.total))+geom_col(fill="lightblue")+theme_minimal()+labs(x="S
peed Limit",y="Number of
casualties")+geom_text(aes(label=cas.total,vjust="bottom"),size = 3)
```

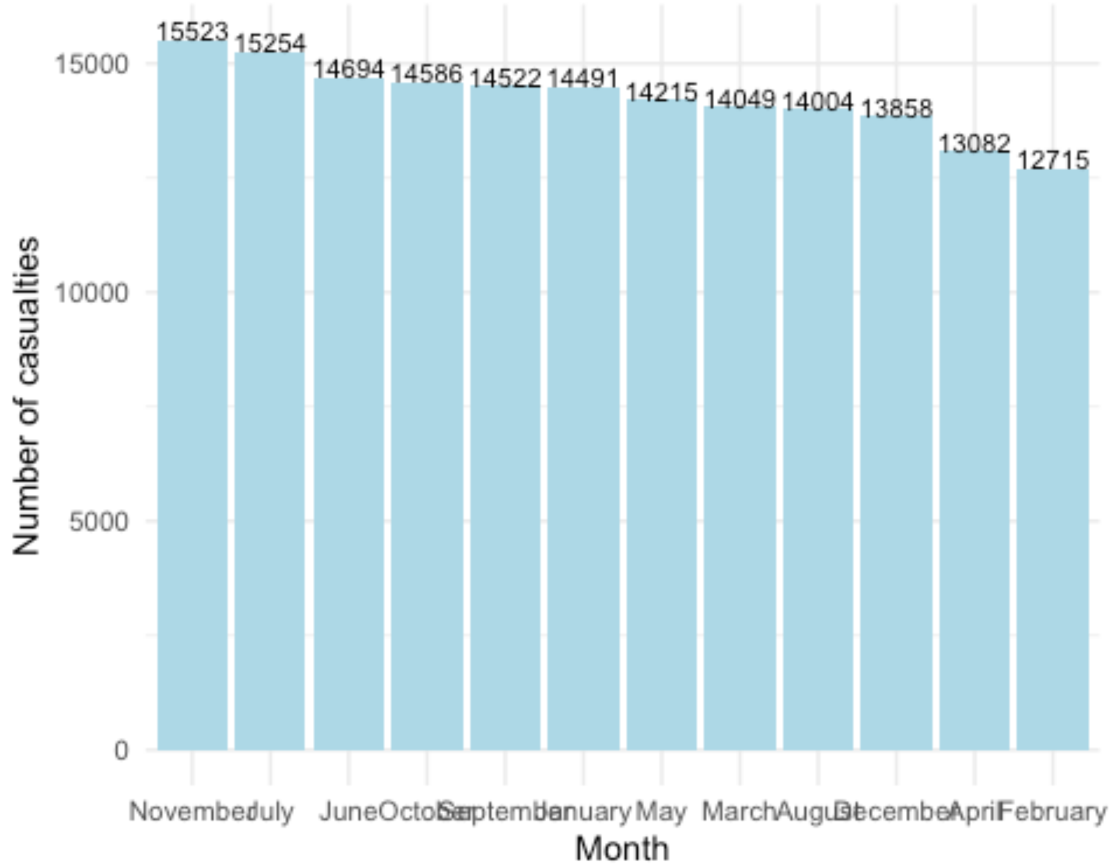



```
oneway.test(formula =
Number_of_Casualties~Road_Surface_Conditions,data=Accident.cleaned,var.equal
= TRUE)

##
##  One-way analysis of means
##
## data:  Number_of_Casualties and Road_Surface_Conditions
## F = 23.81, num df = 4, denom df = 128040, p-value < 2.2e-16

#Question 5: Uni-variate and Bi-variate analysis to find exploratory factors
for accidents


#-----------------------Uni-variate analysis of month----------------------
--

#Analyse accidents in various months
Accident.cleaned<-Accident.cleaned%>%
  mutate(month=month(Accident.cleaned$Date))%>%

mutate(month=recode_factor(.x=month,"1"="January","2"="February","3"="March",
"4"="April","5"="May","6"="June","7"="July","8"="August","9"="September","10"
```

```
="October","11"="November","12"="December"))

month.plot.data<-Accident.cleaned%>%
  drop_na(month)%>%
  group_by(month)%>%
  summarise(No_of_acc=n(),
            cas.total=sum(Number_of_Casualties),
            cas.stat=mean(Number_of_Casualties),
            veh.total=sum(Number_of_Vehicles),
            veh.stat=mean(Number_of_Vehicles))
month.plot.data

## # A tibble: 12 × 6
##    month     No_of_acc cas.total cas.stat veh.total veh.stat
##    <fct>         <int>     <int>    <dbl>     <int>    <dbl>
##  1 January       11147     14491     1.30     19964     1.79
##  2 February       9740     12715     1.31     17689     1.82
##  3 March         10806     14049     1.30     19811     1.83
##  4 April          9773     13082     1.34     18036     1.85
##  5 May           10922     14215     1.30     20247     1.85
##  6 June          11130     14694     1.32     20821     1.87
##  7 July          11450     15254     1.33     21210     1.85
##  8 August        10301     14004     1.36     19279     1.87
##  9 September     11091     14522     1.31     20452     1.84
## 10 October       11194     14586     1.30     20741     1.85
## 11 November      11958     15523     1.30     21902     1.83
## 12 December      10470     13858     1.32     18774     1.79

month.plot.data%>%
  mutate(perc=100*No_of_acc/sum(No_of_acc))%>%
    ggplot(aes(x=reorder(x=month,X=-
perc),y=perc))+geom_col(fill="lightblue")+theme_minimal()+labs(x="Month",y="P
ercentage of accidents")+geom_text(aes(label=sprintf("%0.2f", round(perc,
digits = 2)),vjust="bottom"),size = 3)+ylim(0,10)
```

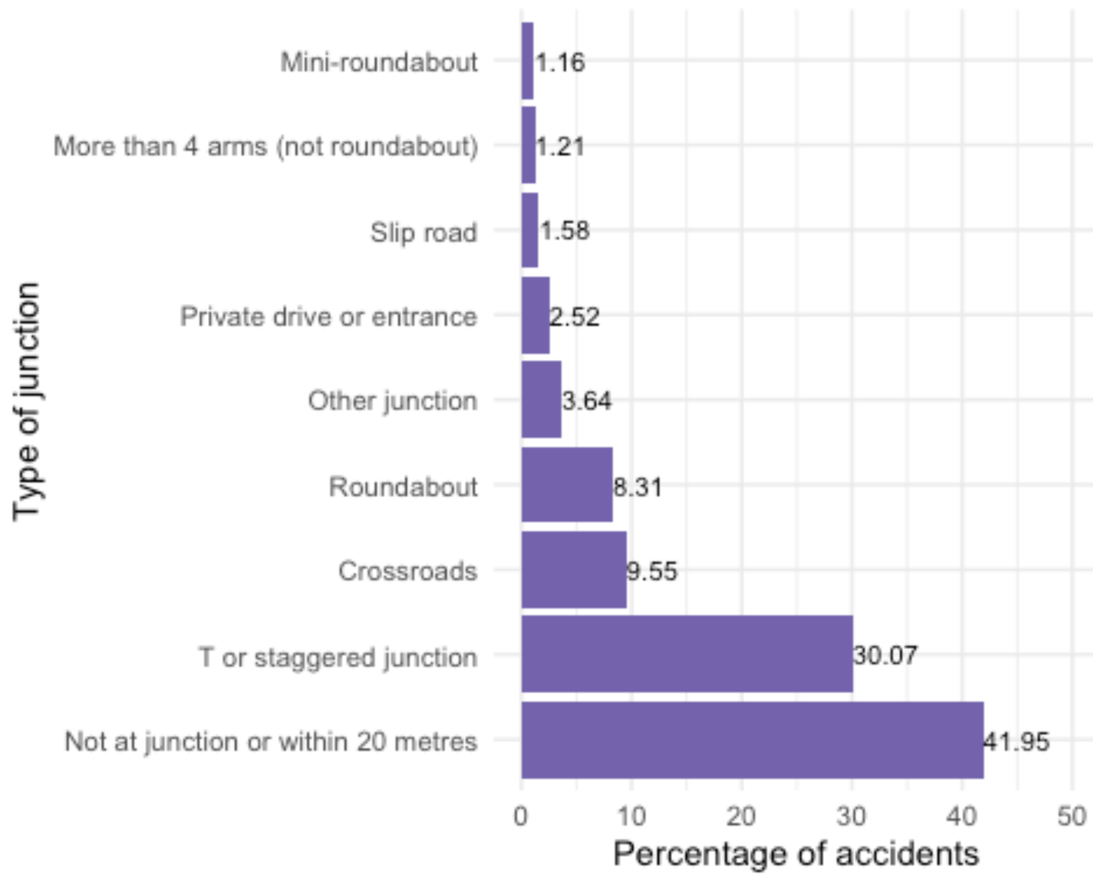



```
month.plot.data%>%
  ggplot(aes(x=reorder(x=month,X=-
cas.total),y=cas.total))+geom_col(fill="lightblue")+theme_minimal()+labs(x="M
onth",y="Number of
casualties")+geom_text(aes(label=cas.total,vjust="bottom"),size = 3)
```

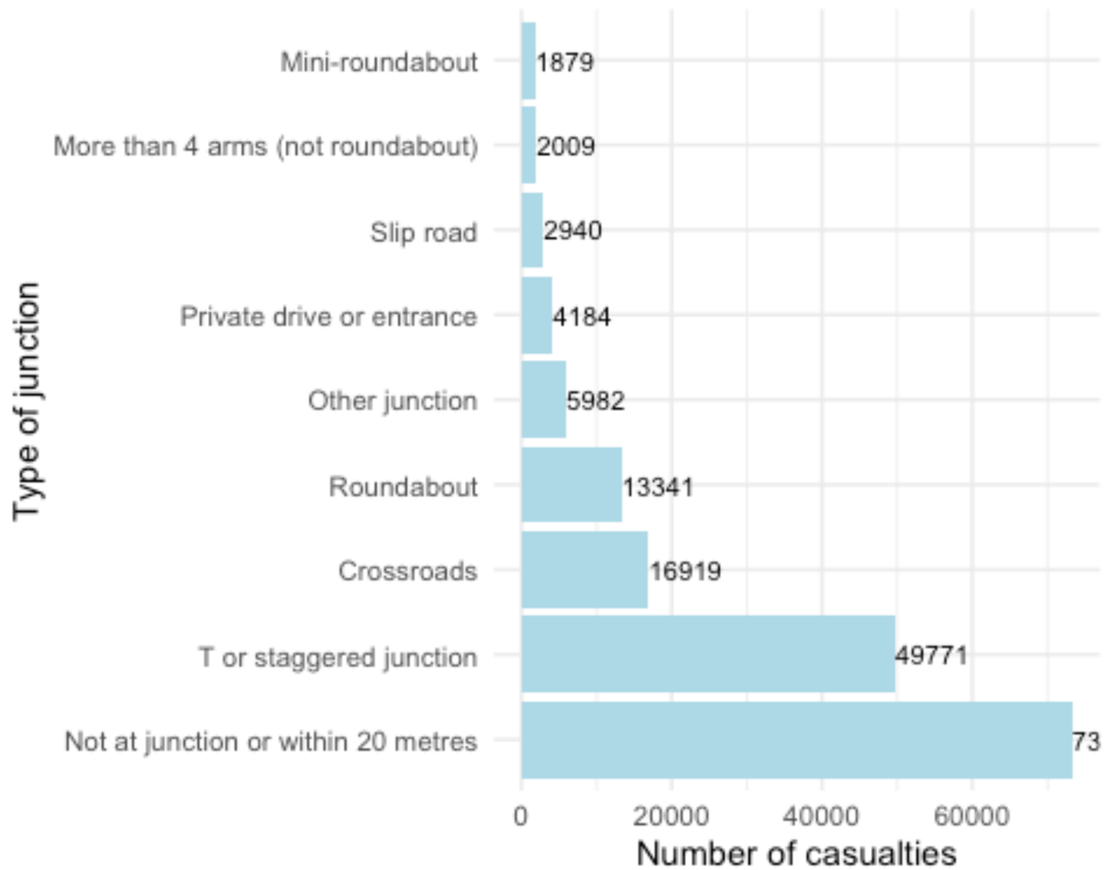



```
oneway.test(formula =
Number_of_Casualties~month,data=Accident.cleaned,var.equal = TRUE)

##
##  One-way analysis of means
##
## data:  Number_of_Casualties and month
## F = 6.738, num df = 11, denom df = 129970, p-value = 2.017e-11

#----------------------Uni-variate analysis of junction detail---------------
-

  junc.detail.plot.data<-Accident.cleaned%>%
  drop_na(Junction_Detail)%>%
  group_by(Junction_Detail)%>%
  summarise(No_of_acc=n(),
            cas.total=sum(Number_of_Casualties),
            cas.stat=mean(Number_of_Casualties),
            veh.total=sum(Number_of_Vehicles),
            veh.stat=mean(Number_of_Vehicles))
junc.detail.plot.data

## # A tibble: 9 × 6
##   Junction_Detail                No_of_acc cas.total cas.stat veh.total
```

```
veh.stat
##   <fct>                                  <int>     <int>    <dbl>     <int>
<dbl>
## 1 Not at junction or within 20 …        54278     73318     1.35     97981
1.81
## 2 Roundabout                            10751     13341     1.24     20275
1.89
## 3 Mini-roundabout                        1501      1879     1.25      2773
1.85
## 4 T or staggered junction               38902     49771     1.28     71830
1.85
## 5 Slip road                              2048      2940     1.44      4145
2.02
## 6 Crossroads                            12352     16919     1.37     23338
1.89
## 7 More than 4 arms (not roundab…         1567      2009     1.28      2817
1.80
## 8 Private drive or entrance              3264      4184     1.28      6275
1.92
## 9 Other junction                         4710      5982     1.27      8512
1.81

junc.detail.plot.data%>%
  mutate(perc=100*No_of_acc/sum(No_of_acc))%>%
    ggplot(aes(x=reorder(x=Junction_Detail,X=-
perc),y=perc))+geom_col(fill="#7463AC")+theme_minimal()+labs(x="Type of
junction",y="Percentage of accidents")+geom_text(aes(label=sprintf("%0.2f",
round(perc, digits = 2)),hjust="left"),size = 3)+ylim(0,50)+coord_flip()
```




```
junc.detail.plot.data%>%
  ggplot(aes(x=reorder(x=Junction_Detail,X=-
cas.total),y=cas.total))+geom_col(fill="lightblue")+theme_minimal()+labs(x="T
ype of junction",y="Number of
casualties")+geom_text(aes(label=cas.total,hjust="left"),size =
3)+coord_flip()
```

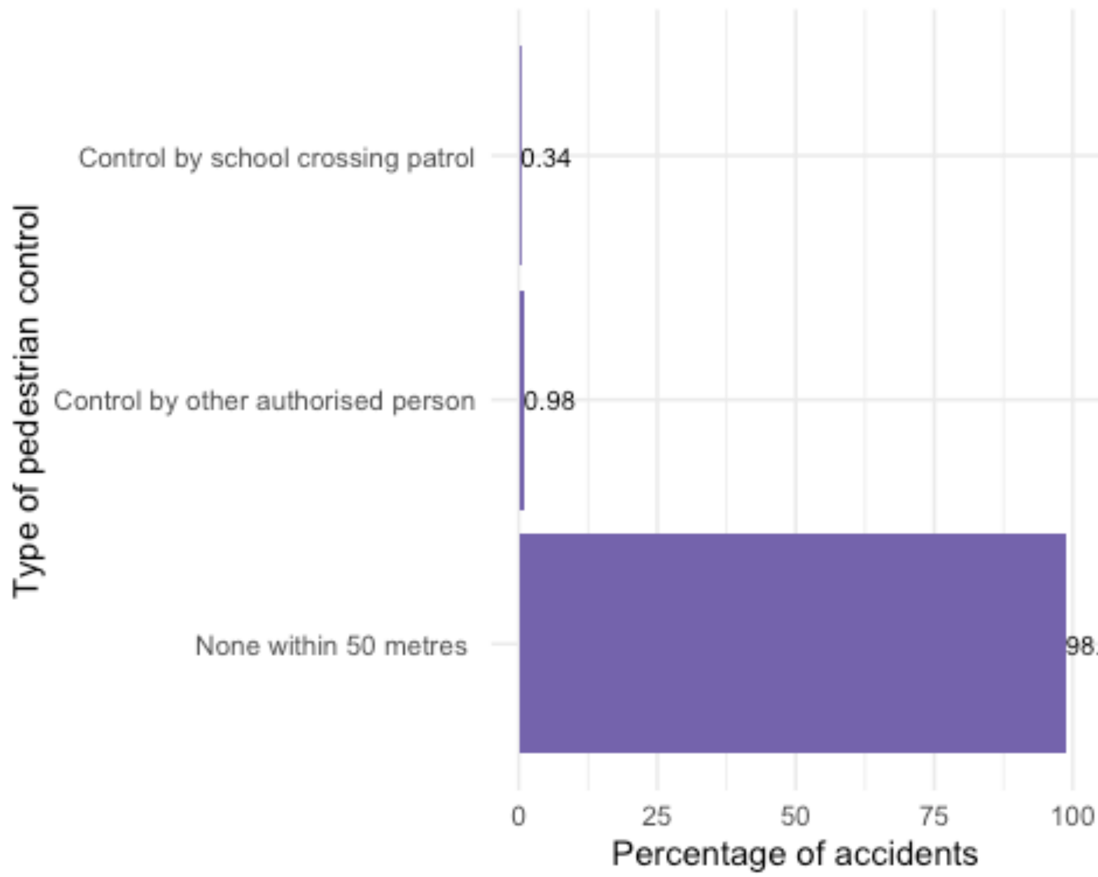



```
oneway.test(formula =
Number_of_Casualties~Junction_Detail,data=Accident.cleaned,var.equal = TRUE)

##
##  One-way analysis of means
##
## data:  Number_of_Casualties and Junction_Detail
## F = 56.535, num df = 8, denom df = 129364, p-value < 2.2e-16

#--------Uni-variate analysis of junction detail dropped due to more NA's----
---

  junc.control.plot.data<-Accident.cleaned%>%
  drop_na(Junction_Control)%>%
  group_by(Junction_Control)%>%
  summarise(No_of_acc=n(),
            cas.total=sum(Number_of_Casualties),
            cas.stat=mean(Number_of_Casualties),
            veh.total=sum(Number_of_Vehicles),
            veh.stat=mean(Number_of_Vehicles))
#------------------Uni-variate analysis of pedestrian control-------------

  ped.human.plot.data<-Accident.cleaned%>%
```

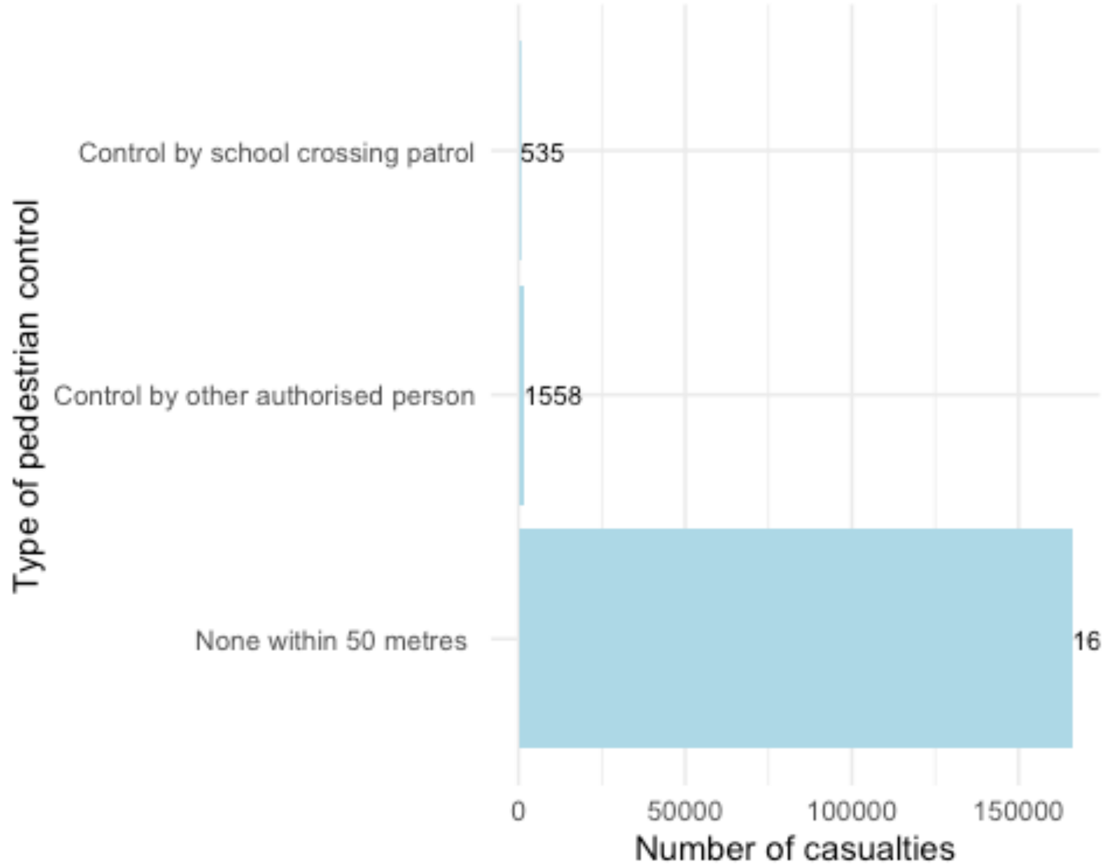
```
  drop_na(Pedestrian_Crossing_Human_Control)%>%
  group_by(Pedestrian_Crossing_Human_Control)%>%
  summarise(No_of_acc=n(),
            cas.total=sum(Number_of_Casualties),
            cas.stat=mean(Number_of_Casualties),
            veh.total=sum(Number_of_Vehicles),
            veh.stat=mean(Number_of_Vehicles))
ped.human.plot.data
```

```
## # A tibble: 3 × 6
##   Pedestrian_Crossing_Human_Con… No_of_acc cas.total cas.stat veh.total
veh.stat
##   <fct>                              <int>     <int>    <dbl>     <int>
<dbl>
## 1 "None within 50 metres "          125724    166089     1.32    231803
1.84
## 2 "Control by school crossing p…       438       535     1.22       724
1.65
## 3 "Control by other authorised …      1246      1558     1.25      2079
1.67
```

```
ped.human.plot.data%>%
  mutate(perc=100*No_of_acc/sum(No_of_acc))%>%
    ggplot(aes(x=reorder(x=Pedestrian_Crossing_Human_Control,X=-
perc),y=perc))+geom_col(fill="#7463AC")+theme_minimal()+labs(x="Type of
pedestrian control",y="Percentage of
accidents")+geom_text(aes(label=sprintf("%0.2f", round(perc, digits =
2)),hjust="left"),size = 3)+ylim(0,100)+coord_flip()
```



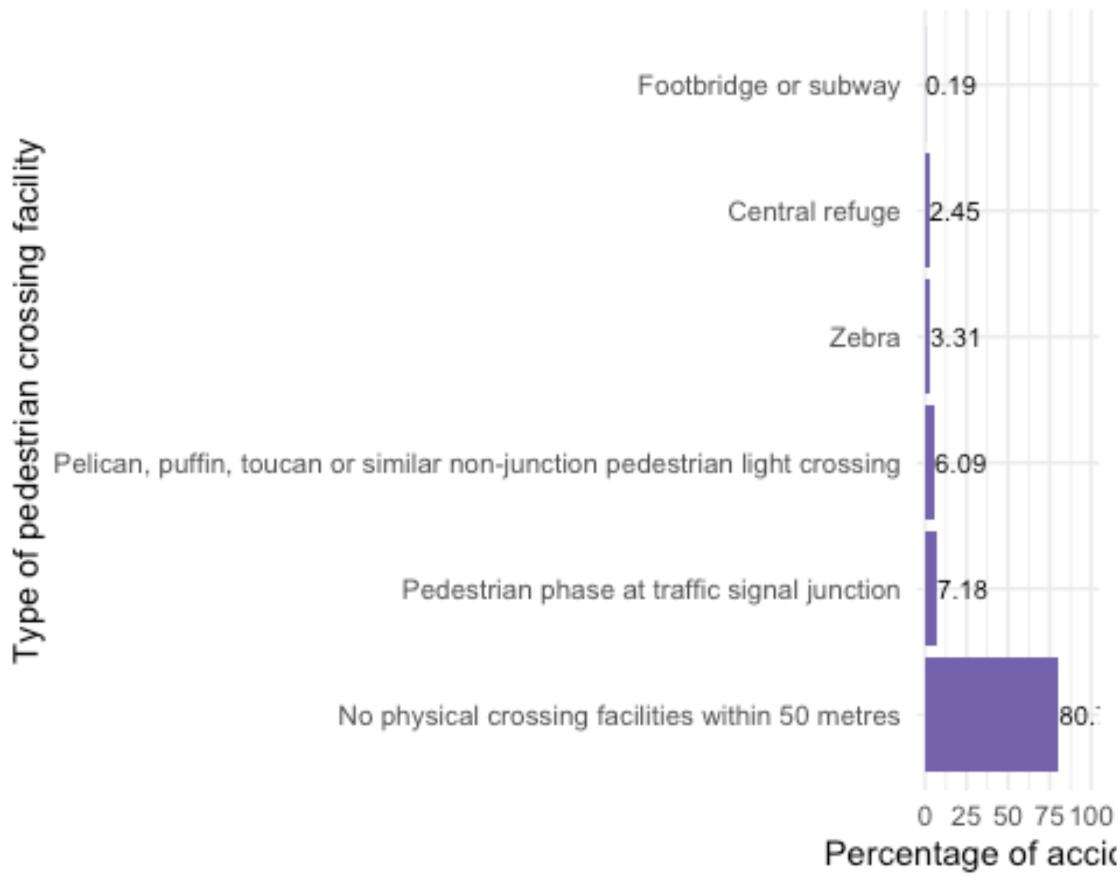

```
ped.human.plot.data%>%
  ggplot(aes(x=reorder(x=Pedestrian_Crossing_Human_Control,X=-
cas.total),y=cas.total))+geom_col(fill="lightblue")+theme_minimal()+labs(x="T
ype of pedestrian control",y="Number of
casualties")+geom_text(aes(label=cas.total,hjust="left"),size =
3)+coord_flip()
```



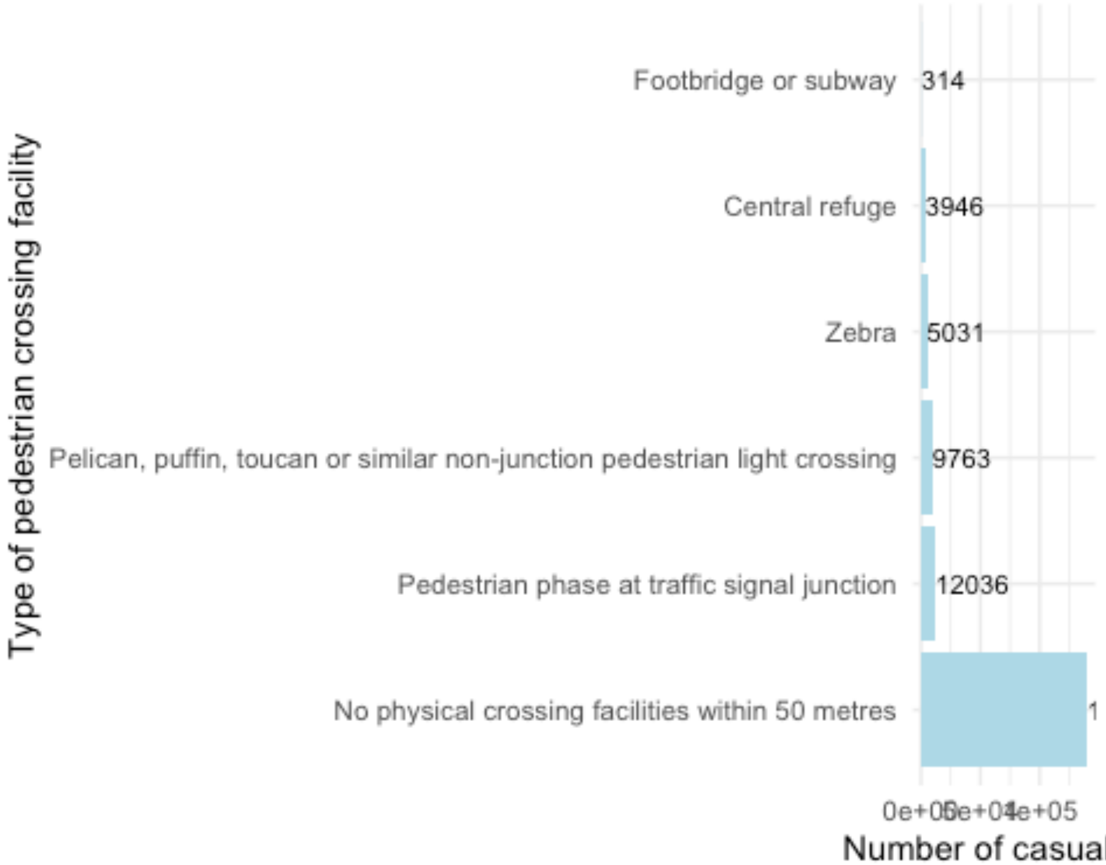

```
oneway.test(formula =
Number_of_Casualties~Pedestrian_Crossing_Human_Control,data=Accident.cleaned,
var.equal = TRUE)

##
##  One-way analysis of means
##
## data:  Number_of_Casualties and Pedestrian_Crossing_Human_Control
## F = 8.7893, num df = 2, denom df = 127405, p-value = 0.0001525

#---------------Uni-variate analysis of pedestrian facility-----------------
-

  ped.fac.plot.data<-Accident.cleaned%>%
  drop_na(Pedestrian_Crossing_Physical_Facilities)%>%
  group_by(Pedestrian_Crossing_Physical_Facilities)%>%
  summarise(No_of_acc=n(),
            cas.total=sum(Number_of_Casualties),
            cas.stat=mean(Number_of_Casualties),
            veh.total=sum(Number_of_Vehicles),
            veh.stat=mean(Number_of_Vehicles))
ped.fac.plot.data
```

```
## # A tibble: 6 × 6
##   Pedestrian_Crossing_Physical_… No_of_acc cas.total cas.stat veh.total
veh.stat
##   <fct>                              <int>     <int>    <dbl>     <int>
<dbl>
## 1 No physical crossing faciliti…    105007    139903     1.33    196165
1.87
## 2 Zebra                               4298      5031     1.17      6884
1.60
## 3 Pelican, puffin, toucan or si…      7915      9763     1.23     13418
1.70
## 4 Pedestrian phase at traffic s…      9338     12036     1.29     16530
1.77
## 5 Footbridge or subway                 241       314     1.30       431
1.79
## 6 Central refuge                      3183      3946     1.24      5498
1.73
```

```
ped.fac.plot.data%>%
  mutate(perc=100*No_of_acc/sum(No_of_acc))%>%
    ggplot(aes(x=reorder(x=Pedestrian_Crossing_Physical_Facilities,X=-
perc),y=perc))+geom_col(fill="#7463AC")+theme_minimal()+labs(x="Type of
pedestrian crossing facility",y="Percentage of
accidents")+geom_text(aes(label=sprintf("%0.2f", round(perc, digits =
2)),hjust="left"),size = 3)+ylim(0,100)+coord_flip()
```



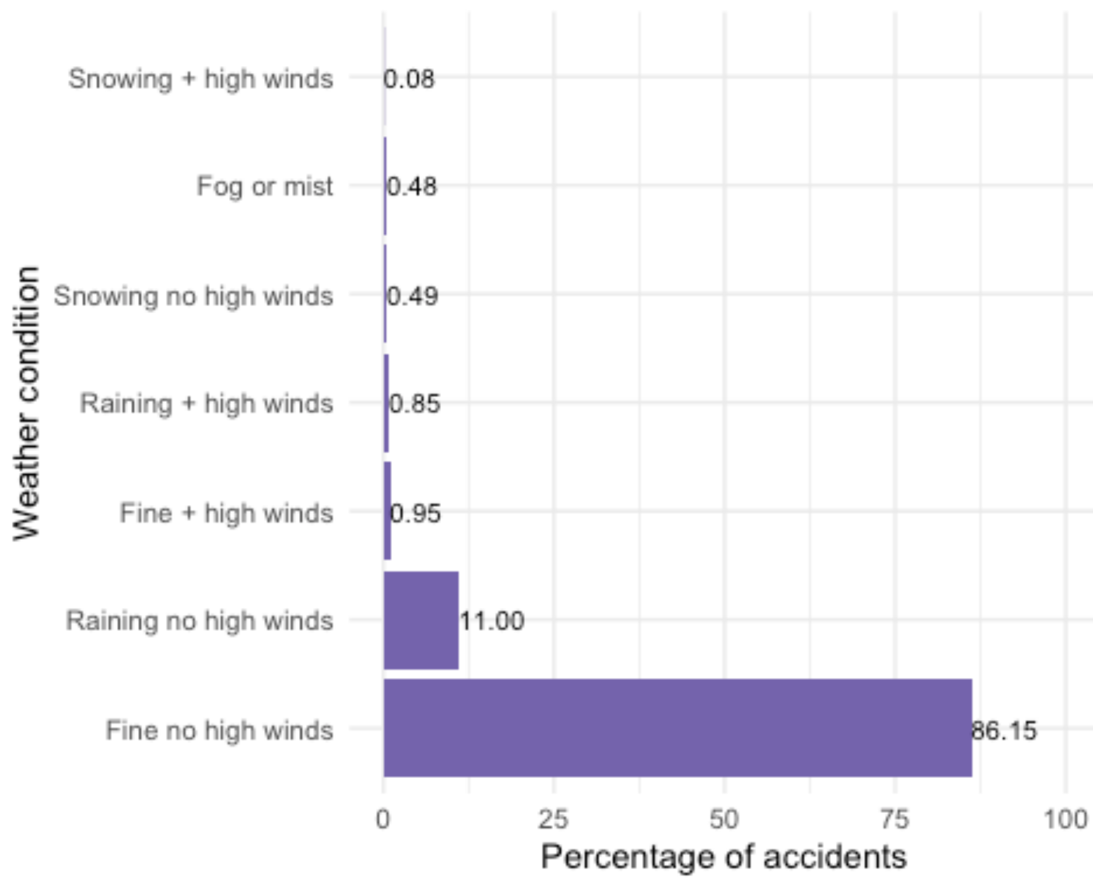

```
ped.fac.plot.data%>%
  ggplot(aes(x=reorder(x=Pedestrian_Crossing_Physical_Facilities,X=-
cas.total),y=cas.total))+geom_col(fill="lightblue")+theme_minimal()+labs(x="T
ype of pedestrian crossing facility",y="Number of
casualties")+geom_text(aes(label=cas.total,hjust="left"),size =
3)+coord_flip()
```



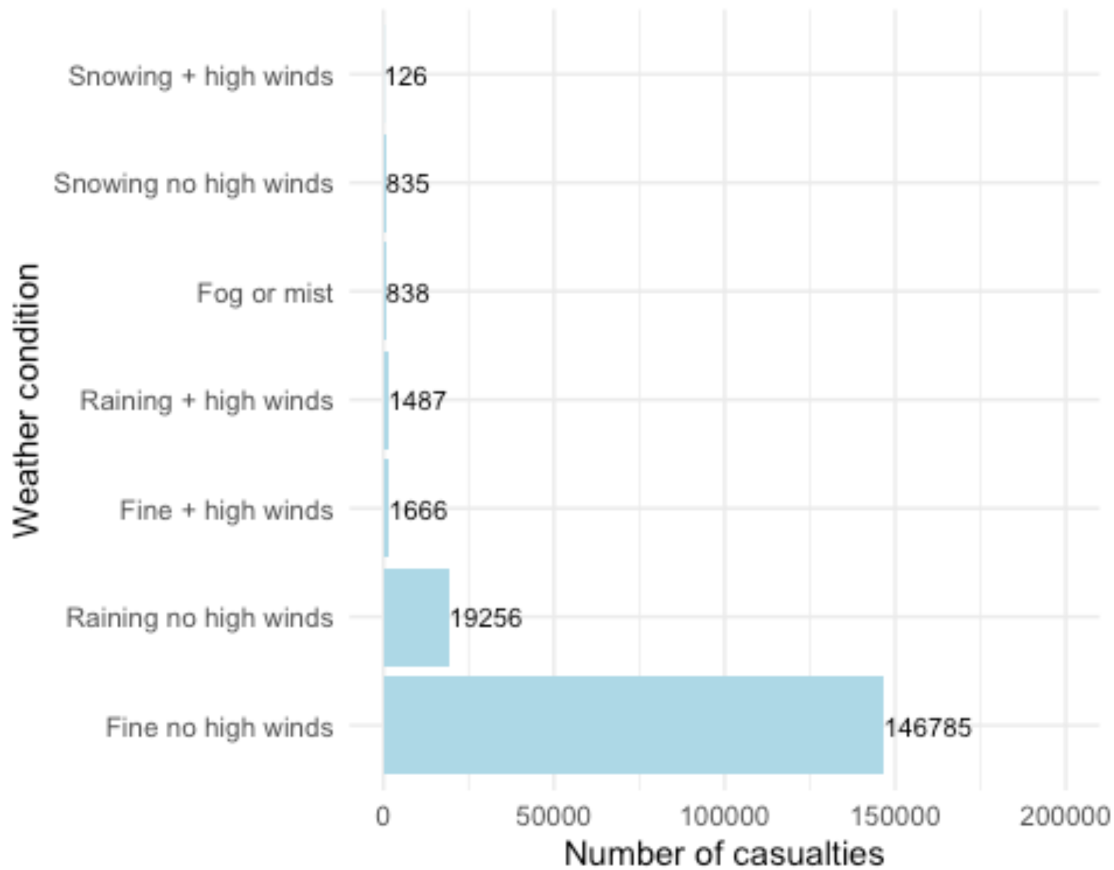

```
oneway.test(formula =
Number_of_Casualties~Pedestrian_Crossing_Physical_Facilities,data=Accident.cl
eaned,var.equal = TRUE)

##
##  One-way analysis of means
##
## data:  Number_of_Casualties and Pedestrian_Crossing_Physical_Facilities
## F = 67.819, num df = 5, denom df = 129976, p-value < 2.2e-16

#------------------Uni-variate analysis of weather-----------------------

  weather.plot.data<-Accident.cleaned%>%
  drop_na(Weather_Conditions)%>%
  group_by(Weather_Conditions)%>%
  summarise(No_of_acc=n(),
            cas.total=sum(Number_of_Casualties),
            cas.stat=mean(Number_of_Casualties),
            veh.total=sum(Number_of_Vehicles),
            veh.stat=mean(Number_of_Vehicles))
weather.plot.data

## # A tibble: 7 × 6
##   Weather_Conditions    No_of_acc cas.total cas.stat veh.total veh.stat
```

```
##   <fct>                     <int>     <int>    <dbl>     <int>    <dbl>
## 1 Fine no high winds        111975    146785     1.31    206627     1.85
## 2 Raining no high winds      14300     19256     1.35     25717     1.80
## 3 Snowing no high winds        641       835     1.30      1095     1.71
## 4 Fine + high winds           1241      1666     1.34      2262     1.82
## 5 Raining + high winds        1102      1487     1.35      1917     1.74
## 6 Snowing + high winds          99       126     1.27       173     1.75
## 7 Fog or mist                  624       838     1.34      1135     1.82
```

```
weather.plot.data%>%
  mutate(perc=100*No_of_acc/sum(No_of_acc))%>%
    ggplot(aes(x=reorder(x=Weather_Conditions,X=-
perc),y=perc))+geom_col(fill="#7463AC")+theme_minimal()+labs(x="Weather
condition",y="Percentage of accidents")+geom_text(aes(label=sprintf("%0.2f",
round(perc, digits = 2)),hjust="left"),size = 3)+ylim(0,100)+coord_flip()
```



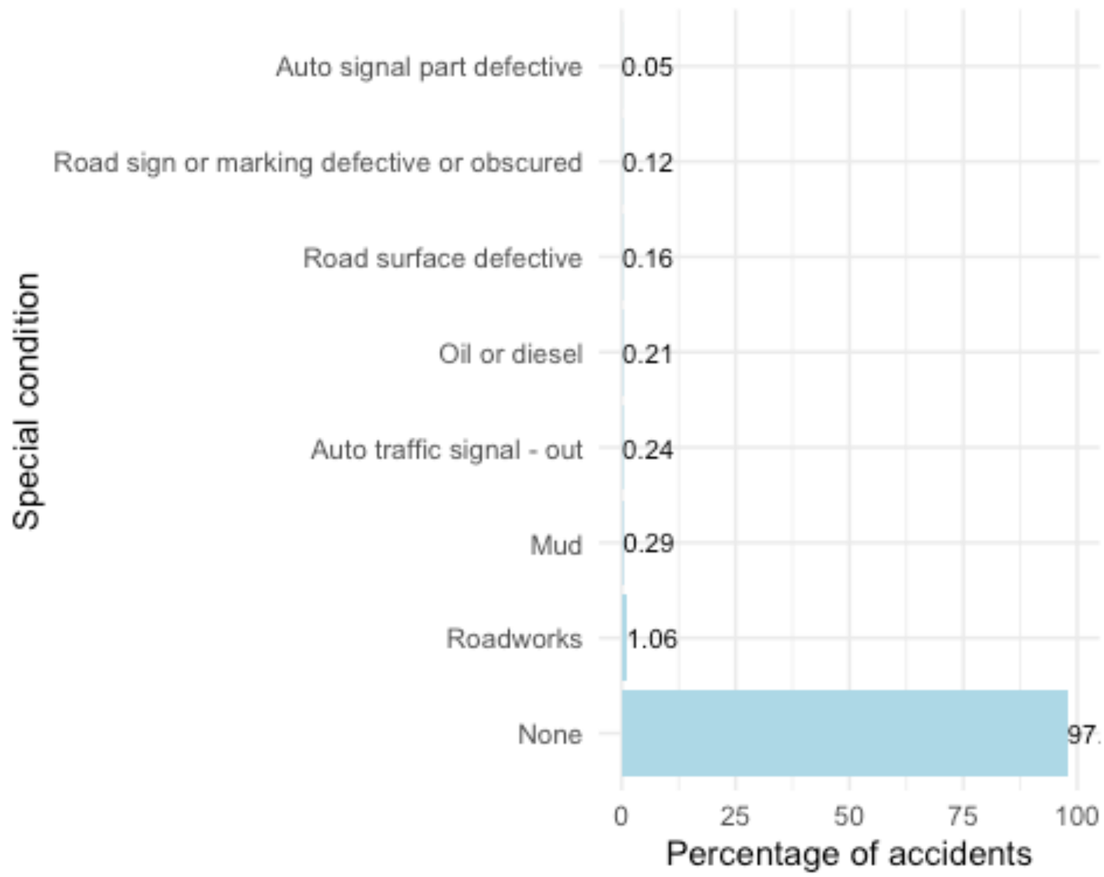

```
weather.plot.data%>%
  ggplot(aes(x=reorder(x=Weather_Conditions,X=-
cas.total),y=cas.total))+geom_col(fill="lightblue")+theme_minimal()+labs(x="W
eather condition",y="Number of
casualties")+geom_text(aes(label=cas.total,hjust="left"),size =
3)+coord_flip()+ylim(0,200000)
```



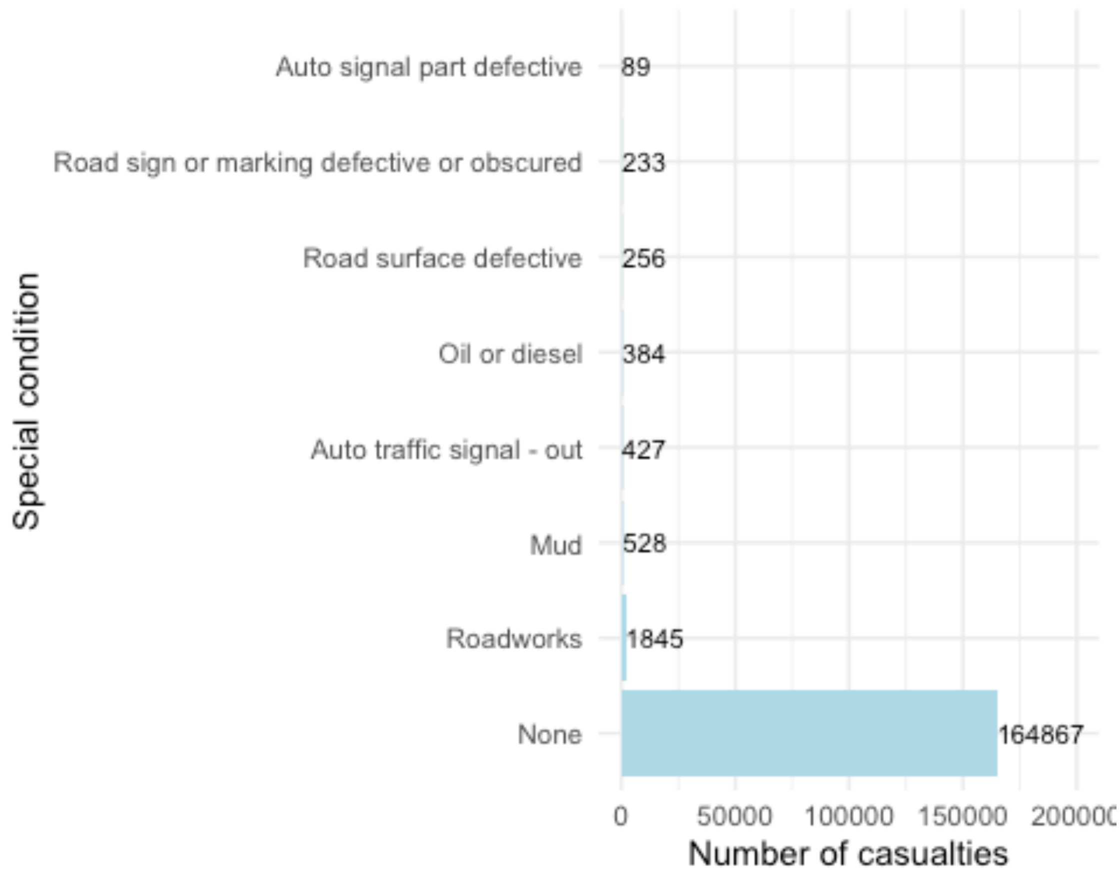

```
oneway.test(formula =
Number_of_Casualties~Weather_Conditions,data=Accident.cleaned,var.equal =
TRUE)

##
##  One-way analysis of means
##
## data:  Number_of_Casualties and Weather_Conditions
## F = 5.4419, num df = 6, denom df = 129975, p-value = 1.225e-05

#----------------------Uni-variate analysis of special conditions------------
-

  spec.cond.plot.data<-Accident.cleaned%>%
  drop_na(Special_Conditions_at_Site)%>%
  group_by(Special_Conditions_at_Site)%>%
  summarise(No_of_acc=n(),
            cas.total=sum(Number_of_Casualties),
            cas.stat=mean(Number_of_Casualties),
            veh.total=sum(Number_of_Vehicles),
            veh.stat=mean(Number_of_Vehicles))
spec.cond.plot.data
```

```
## # A tibble: 8 × 6
##   Special_Conditions_at_Site     No_of_acc cas.total cas.stat veh.total
veh.stat
##   <fct>                              <int>     <int>    <dbl>     <int>
<dbl>
## 1 None                              125040    164867     1.32    230327
1.84
## 2 Auto traffic signal - out            308       427     1.39       601
1.95
## 3 Auto signal part defective            64        89     1.39       127
1.98
## 4 Road sign or marking defectiv…       159       233     1.47       301
1.89
## 5 Roadworks                           1352      1845     1.36      2666
1.97
## 6 Road surface defective               208       256     1.23       293
1.41
## 7 Oil or diesel                        270       384     1.42       420
1.56
## 8 Mud                                  375       528     1.41       568
1.51
```

```
spec.cond.plot.data%>%
  mutate(perc=100*No_of_acc/sum(No_of_acc))%>%
    ggplot(aes(x=reorder(x=Special_Conditions_at_Site,X=-
perc),y=perc))+geom_col(fill="lightblue")+theme_minimal()+labs(x="Special
condition",y="Percentage of accidents")+geom_text(aes(label=sprintf("%0.2f",
round(perc, digits = 2)),hjust="left"),size = 3)+ylim(0,100)+coord_flip()
```



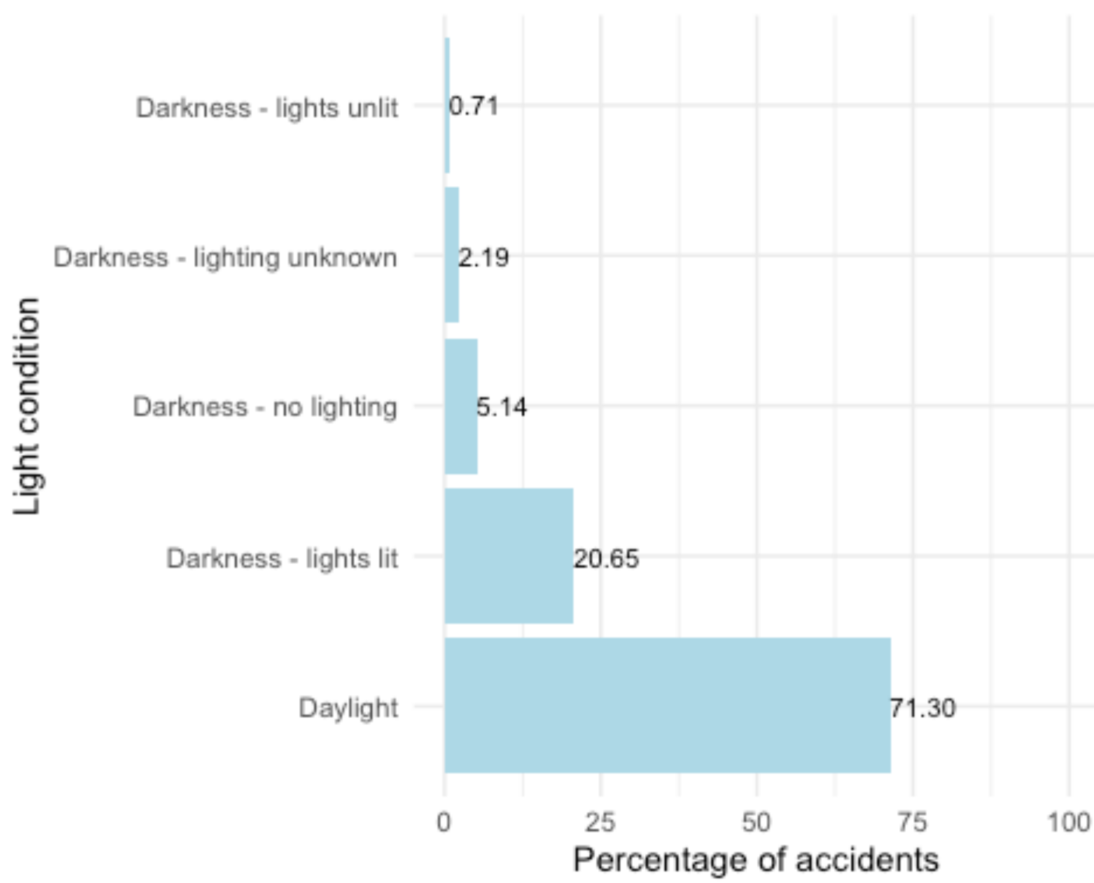

```
spec.cond.plot.data%>%
  ggplot(aes(x=reorder(x=Special_Conditions_at_Site,X=-
cas.total),y=cas.total))+geom_col(fill="lightblue")+theme_minimal()+labs(x="S
pecial condition",y="Number of
casualties")+geom_text(aes(label=cas.total,hjust="left"),size =
3)+coord_flip()+ylim(0,200000)
```




```
oneway.test(formula =
Number_of_Casualties~Special_Conditions_at_Site,data=Accident.cleaned,var.equ
al = TRUE)

##
##  One-way analysis of means
##
## data:  Number_of_Casualties and Special_Conditions_at_Site
## F = 3.7051, num df = 7, denom df = 127768, p-value = 0.0005176

  #--------------------Uni-variate analysis of light conditions--------------

    light.plot.data<-Accident.cleaned%>%
  drop_na(Light_Conditions)%>%
  group_by(Light_Conditions)%>%
  summarise(No_of_acc=n(),
            cas.total=sum(Number_of_Casualties),
            cas.stat=mean(Number_of_Casualties),
            veh.total=sum(Number_of_Vehicles),
            veh.stat=mean(Number_of_Vehicles))
light.plot.data

## # A tibble: 5 × 6
##   Light_Conditions            No_of_acc cas.total cas.stat veh.total
```

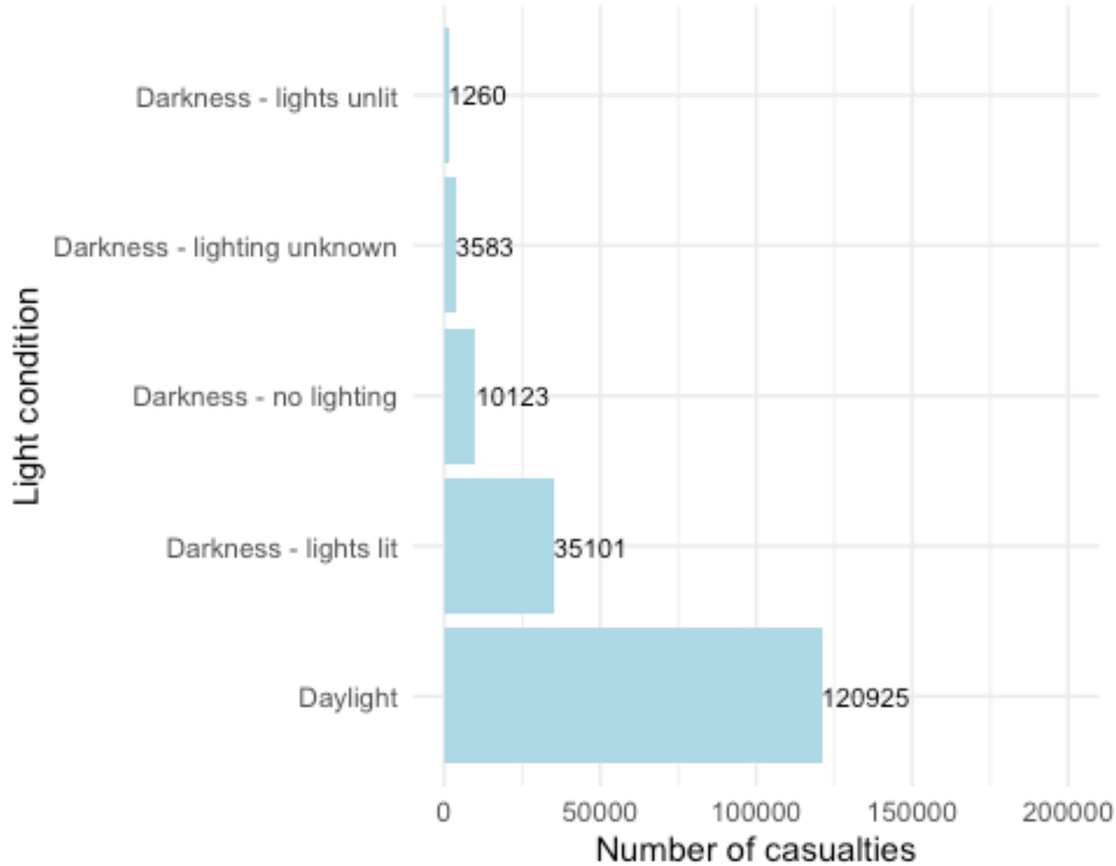
```
veh.stat
##   <fct>                            <int>     <int>    <dbl>     <int>
<dbl>
## 1 Daylight                         92678    120925     1.30    172624
1.86
## 2 Darkness - lights lit            26847     35101     1.31     48369
1.80
## 3 Darkness - lights unlit            929      1260     1.36      1654
1.78
## 4 Darkness - no lighting            6678     10123     1.52     11114
1.66
## 5 Darkness - lighting unknown       2849      3583     1.26      5163
1.81
```

```
light.plot.data%>%
  mutate(perc=100*No_of_acc/sum(No_of_acc))%>%
    ggplot(aes(x=reorder(x=Light_Conditions,X=-
perc),y=perc))+geom_col(fill="lightblue")+theme_minimal()+labs(x="Light
condition",y="Percentage of accidents")+geom_text(aes(label=sprintf("%0.2f",
round(perc, digits = 2)),hjust="left"),size = 3)+ylim(0,100)+coord_flip()
```

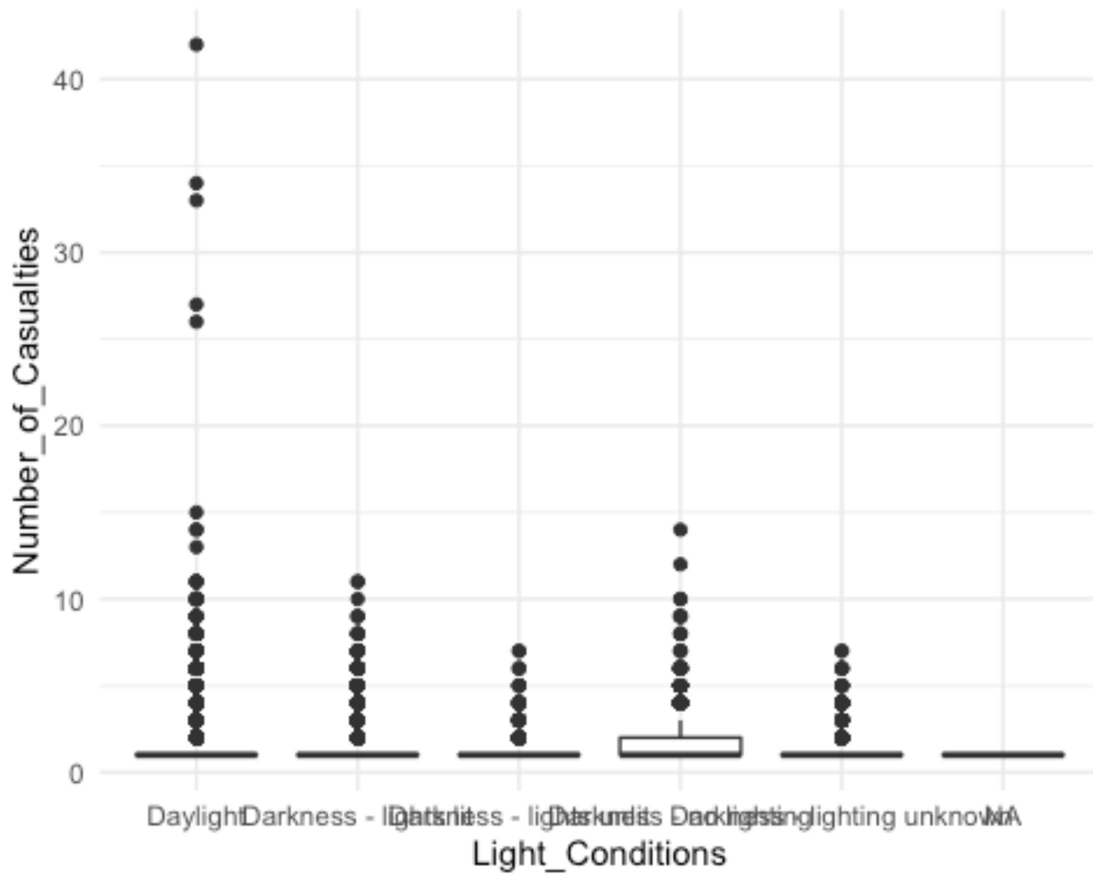



```
light.plot.data%>%
  ggplot(aes(x=reorder(x=Light_Conditions,X=-
```

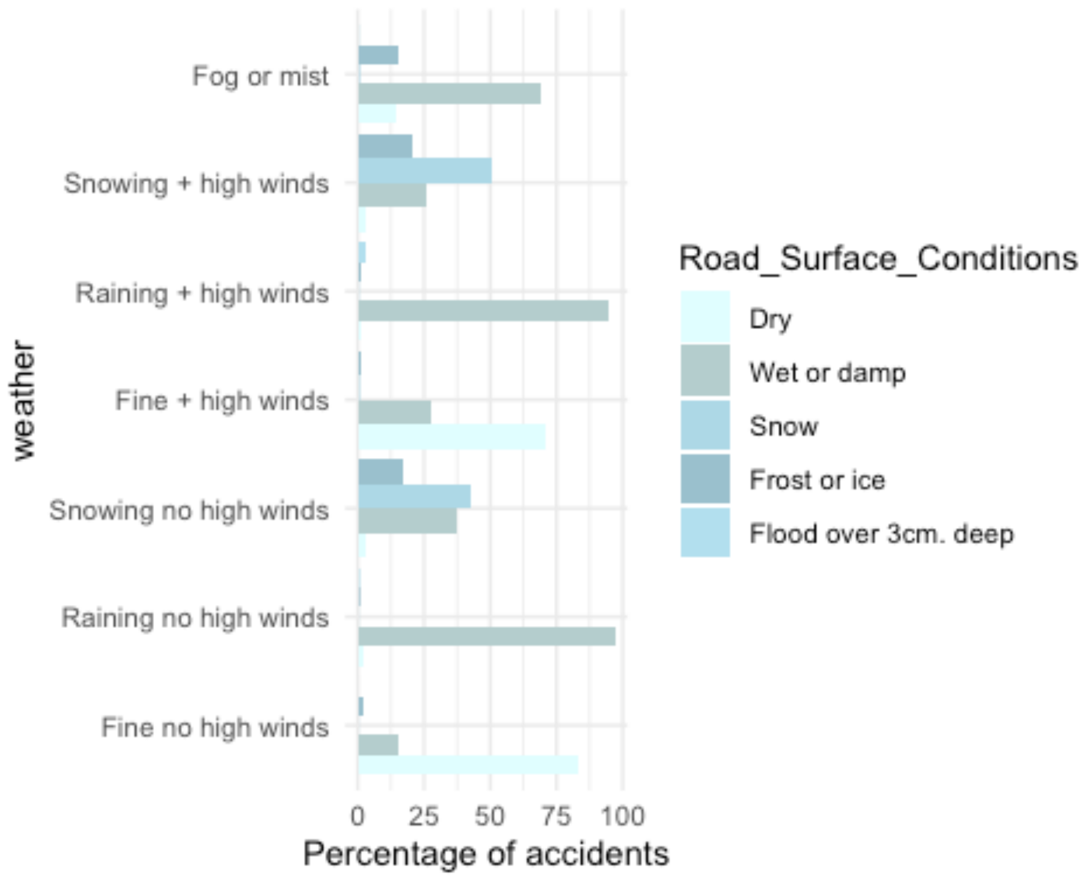
```
cas.total),y=cas.total))+geom_col(fill="lightblue")+theme_minimal()+labs(x="L
ight condition",y="Number of
casualties")+geom_text(aes(label=cas.total,hjust="left"),size =
3)+coord_flip()+ylim(0,200000)
```




```
oneway.test(formula =
Number_of_Casualties~Light_Conditions,data=Accident.cleaned,var.equal = TRUE)

##
##  One-way analysis of means
##
## data:  Number_of_Casualties and Light_Conditions
## F = 124.88, num df = 4, denom df = 129976, p-value < 2.2e-16

Accident.cleaned%>%

ggplot(aes(x=Light_Conditions,y=Number_of_Casualties))+geom_boxplot()+theme_m
inimal()
```




```
#--------------------End of all Uni-variate analysis------------------------

#----------------------Bi-variate analysis of road surface and weather-------
--

road.weather<-Accident.cleaned%>%
  drop_na(Road_Surface_Conditions)%>%
  drop_na(Weather_Conditions)%>%
  group_by(Road_Surface_Conditions,Weather_Conditions)%>%
  count()%>%
  group_by(Weather_Conditions)%>%
  mutate(perc=100*n/sum(n))
road.weather%>%

ggplot(aes(x=Weather_Conditions,y=perc,fill=Road_Surface_Conditions))+geom_co
l(position = "dodge")+theme_minimal()+labs(x="weather","y"="Percentage of
accidents")+coord_flip()+scale_fill_manual(values=c("lightcyan","lightcyan3",
"lightblue","lightblue3","lightblue2"))
```

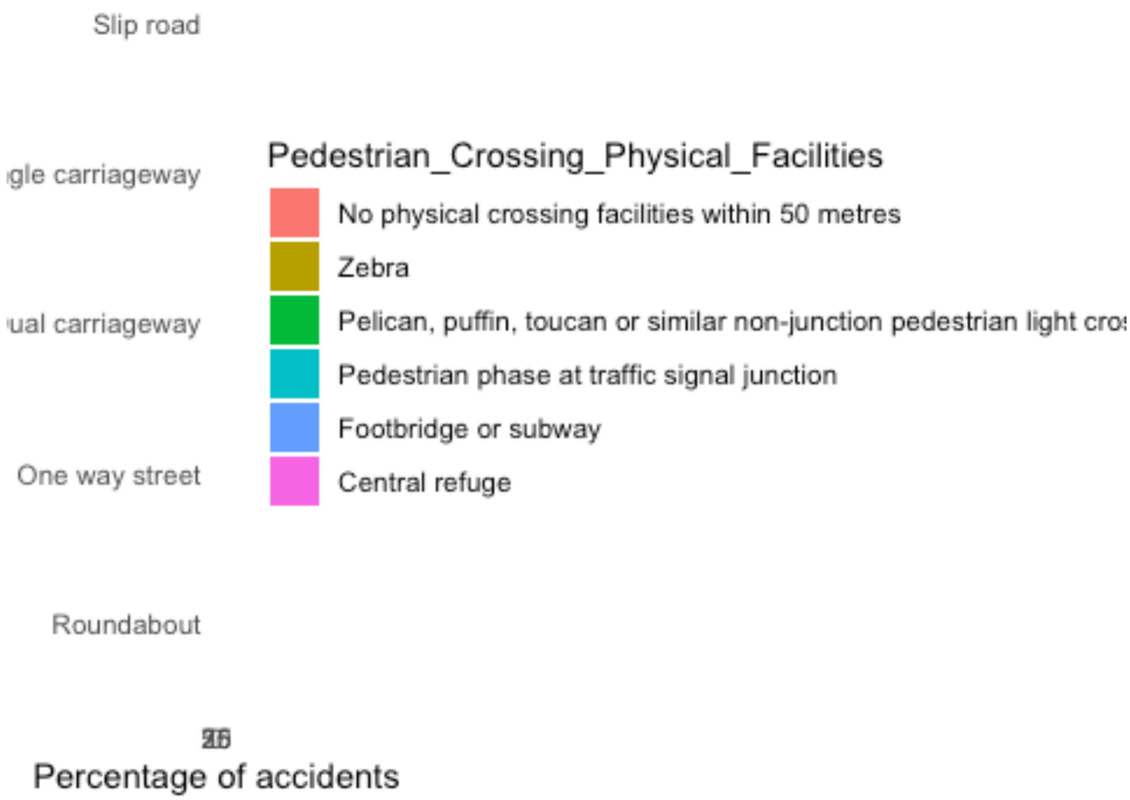



```
chisq.test(x=Accident.cleaned$Weather_Conditions,y=Accident.cleaned$Road_Surf
ace_Conditions)

## Warning in chisq.test(x = Accident.cleaned$Weather_Conditions, y =
## Accident.cleaned$Road_Surface_Conditions): Chi-squared approximation may
be
## incorrect

##
##  Pearson's Chi-squared test
##
## data:  Accident.cleaned$Weather_Conditions and
Accident.cleaned$Road_Surface_Conditions
## X-squared = 94272, df = 24, p-value < 2.2e-16

CrossTable(x=Accident.cleaned$Weather_Conditions,y=Accident.cleaned$Road_Surf
ace_Conditions,expected = TRUE,         prop.c = FALSE,         prop.t = FALSE,
prop.chisq = FALSE,         chisq = TRUE,         sresid = TRUE)

## Warning in chisq.test(tab, correct = FALSE, ...): Chi-squared
approximation may
## be incorrect
```

```
##    Cell Contents
## |-------------------------|
## |                       N |
## |              Expected N |
## |           N / Row Total |
## |            Std Residual |
## |-------------------------|
##
##
=============================================================================
===
##                Accident.cleaned$Road_Surface_Conditions
## Accdn.$W_C         Dry   Wet or dmp      Snow   Frst or ic   Fld o 3. d
Total
## --------------------------------------------------------------------------
------
## Fn n hgh w      91370        16633        81         1979           12
110075
##              79628.5      27993.9     367.9       1993.5         91.1
##                0.830        0.151     0.001        0.018        0.000
0.860
##               41.609      -67.902   -14.959       -0.326       -8.289
## --------------------------------------------------------------------------
------
## Rnng n h w        259        13855        14           83           63
14274
##              10325.8       3630.1      47.7        258.5         11.8
##                0.018        0.971     0.001        0.006        0.004
0.111
##              -99.067      169.706    -4.881      -10.916       14.890
## --------------------------------------------------------------------------
------
## Snwn n h w         17          238       273          109            0
637
##                460.8        162.0       2.1         11.5          0.5
##                0.027        0.374     0.429        0.171        0.000
0.005
##              -20.674        5.971   185.632       28.695       -0.726
## --------------------------------------------------------------------------
------
## Fn + hgh w        875          337         5           18            0
1235
##                893.4        314.1       4.1         22.4          1.0
##                0.709        0.273     0.004        0.015        0.000
0.010
##               -0.616        1.293     0.429       -0.923       -1.011
## --------------------------------------------------------------------------
------
## Rnng + h w         16         1042         2           12           30
1102
```

```
##                   797.2         280.3      3.7         20.0          0.9
##                   0.015         0.946    0.002        0.011        0.027
0.009
##                 -27.668        45.502   -0.877       -1.781       30.454
## -----------------------------------------------------------------------------
------
## Snwn + h w            3            26       50           20            0
99
##                    71.6          25.2      0.3          1.8          0.1
##                   0.030         0.263    0.505        0.202        0.000
0.001
##                  -8.108         0.164   86.343       13.597       -0.286
## -----------------------------------------------------------------------------
------
## Fog or mst           88           433        3           98            1
623
##                   450.7         158.4      2.1         11.3          0.5
##                   0.141         0.695    0.005        0.157        0.002
0.005
##                 -17.084        21.813    0.636       25.816        0.674
## -----------------------------------------------------------------------------
------
## Total             92628         32564      428         2319          106
128045
##
=============================================================================
===
##
## Statistics for All Table Factors
##
## Pearson's Chi-squared test
## ------------------------------------------------------------
## Chi^2 = 94272.22      d.f. = 24      p <2e-16
```

```
#-------------Bi-variate analysis of ped fec and road types---------------

road.ped.fes<-Accident.cleaned%>%
  drop_na(Pedestrian_Crossing_Physical_Facilities)%>%
  drop_na(Road_Type)%>%
  group_by(Pedestrian_Crossing_Physical_Facilities,Road_Type)%>%
  count()%>%
  group_by(Road_Type)%>%
  mutate(perc=100*n/sum(n))
road.ped.fes%>%

ggplot(aes(x=Road_Type,y=perc,fill=Pedestrian_Crossing_Physical_Facilities))+
geom_col(position = "dodge")+theme_minimal()+labs(x="Road
type","y"="Percentage of accidents")+coord_flip()
```




```
ch.ped.road<-
chisq.test(x=Accident.cleaned$Pedestrian_Crossing_Physical_Facilities,y=Accid
ent.cleaned$Road_Type)

## Warning in chisq.test(x =
## Accident.cleaned$Pedestrian_Crossing_Physical_Facilities, : Chi-squared
## approximation may be incorrect

res.ped.road<-
CrossTable(x=Accident.cleaned$Pedestrian_Crossing_Physical_Facilities,y=Accid
ent.cleaned$Road_Type,expected = TRUE,         prop.c = FALSE,         prop.t =
FALSE,         prop.chisq = FALSE,         chisq = TRUE,         sresid = TRUE)

## Warning in chisq.test(tab, correct = FALSE, ...): Chi-squared
approximation may
## be incorrect

road.ped.fes

## # A tibble: 30 × 4
## # Groups:   Road_Type [5]
##    Pedestrian_Crossing_Physical_Facilities         Road_Type             n
perc
##    <fct>                                           <fct>             <int>
```

```
<dbl>
##  1 No physical crossing facilities within 50 metr… Roundabout         6667
79.2
##  2 No physical crossing facilities within 50 metr… One way street     2284
67.5
##  3 No physical crossing facilities within 50 metr… Dual carriageway  15750
77.4
##  4 No physical crossing facilities within 50 metr… Single carriagew… 76653
81.7
##  5 No physical crossing facilities within 50 metr… Slip road          1292
87.5
##  6 Zebra                                           Roundabout          343
4.08
##  7 Zebra                                           One way street      254
7.50
##  8 Zebra                                           Dual carriageway    305
1.50
##  9 Zebra                                           Single carriagew…  3290
3.51
## 10 Zebra                                           Slip road            31
2.10
## # … with 20 more rows
```

```
ch.ped.road
```

```
##
##  Pearson's Chi-squared test
##
## data:  Accident.cleaned$Pedestrian_Crossing_Physical_Facilities and
Accident.cleaned$Road_Type
## X-squared = 2809.9, df = 20, p-value < 2.2e-16
```

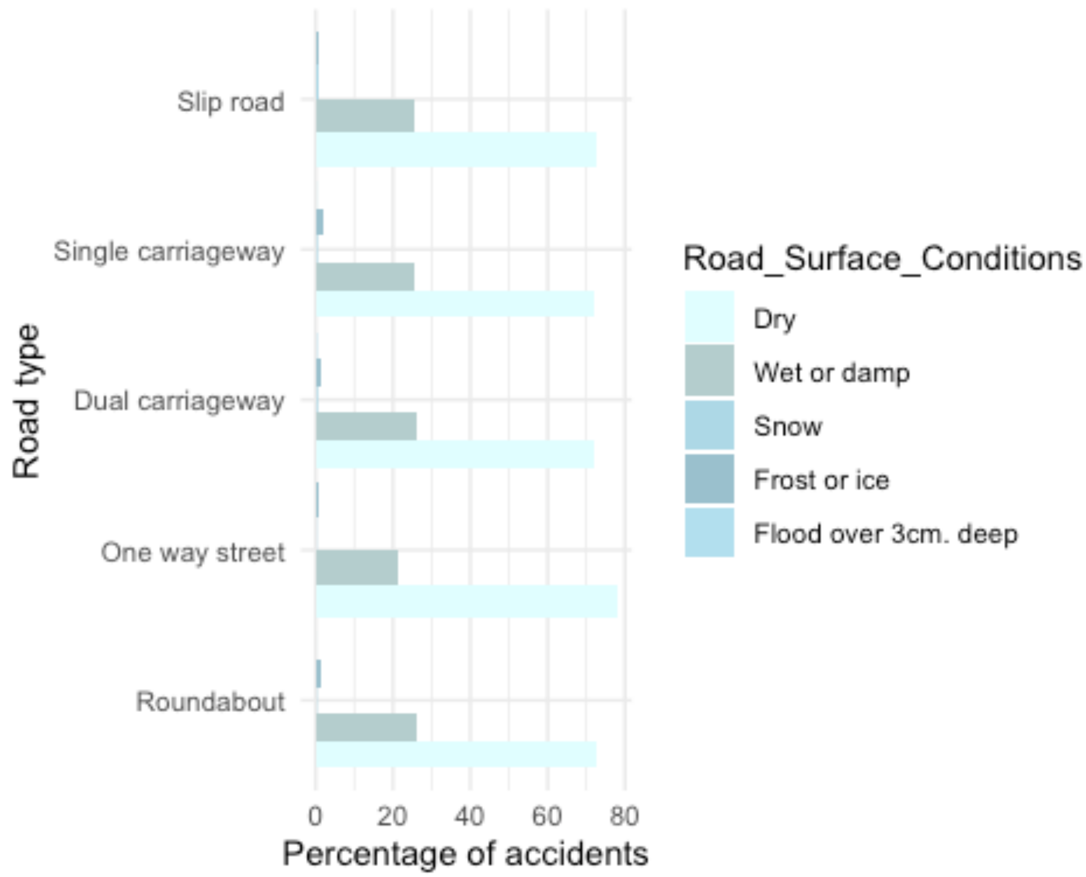
```
res.ped.road
```

```
##    Cell Contents
## |-------------------------|
## |                       N |
## |              Expected N |
## |           N / Row Total |
## |            Std Residual |
## |-------------------------|
##
##
=============================================================================
===
##                Accident.cleaned$Road_Type
## A.$P_C_P_      Roundabot   On wy str   Dl crrgwy   Sngl crrg   Slip road
Total
## -----------------------------------------------------------------------------
------
## N p c f w           6667        2284       15750       76653        1292
```

```
102646
##                   6780.0       2727.5      16384.1      75565.6       1188.9
##                    0.065        0.022        0.153        0.747        0.013
0.806
##                   -1.372       -8.491       -4.954        3.956        2.989
## ----------------------------------------------------------------------------
------
## Zebra                343          254          305         3290           31
4223
##                    278.9        112.2        674.1       3108.9         48.9
##                    0.081        0.060        0.072        0.779        0.007
0.033
##                    3.836       13.385      -14.215        3.248       -2.561
## ----------------------------------------------------------------------------
------
## P, p, t o            433          379         1573         5449           43
7877
##                    520.3        209.3       1257.3       5798.9         91.2
##                    0.055        0.048        0.200        0.692        0.005
0.062
##                   -3.827       11.730        8.903       -4.594       -5.050
## ----------------------------------------------------------------------------
------
## P p a t s            308          399         2342         6141           82
9272
##                    612.4        246.4       1480.0       6825.8        107.4
##                    0.033        0.043        0.253        0.662        0.009
0.073
##                  -12.302        9.724       22.408       -8.289       -2.451
## ----------------------------------------------------------------------------
------
## Ftbrd o s             73            5           81           67           12
238
##                     15.7          6.3         38.0        175.2          2.8
##                    0.307        0.021        0.340        0.282        0.050
0.002
##                   14.447       -0.526        6.978       -8.175        5.567
## ----------------------------------------------------------------------------
------
## Cntrl rfg            593           65          289         2211           16
3174
##                    209.6         84.3        506.6       2336.6         36.8
##                    0.187        0.020        0.091        0.697        0.005
0.025
##                   26.476       -2.106       -9.669       -2.599       -3.425
## ----------------------------------------------------------------------------
------
## Total               8417         3386        20340        93811         1476
127430
##
```

```
=============================================================================
===
##
## Statistics for All Table Factors
##
## Pearson's Chi-squared test
## ------------------------------------------------------------
## Chi^2 = 2809.851      d.f. = 20      p <2e-16
```

```
#--------------------Bi-variate analysis of ped con and road types----------

road.surf<-Accident.cleaned%>%
  drop_na(Road_Surface_Conditions)%>%
  drop_na(Road_Type)%>%
  group_by(Road_Surface_Conditions,Road_Type)%>%
  count()%>%
  group_by(Road_Type)%>%
  mutate(perc=100*n/sum(n))

road.surf%>%

ggplot(aes(x=Road_Type,y=perc,fill=Road_Surface_Conditions))+geom_col(positio
n = "dodge")+theme_minimal()+labs(x="Road type","y"="Percentage of
accidents")+coord_flip()+scale_fill_manual(values=c("lightcyan","lightcyan3",
"lightblue","lightblue3","lightblue2"))
```




```
chisq.test(x=Accident.cleaned$Road_Surface_Conditions,y=Accident.cleaned$Road
_Type)

## Warning in chisq.test(x = Accident.cleaned$Road_Surface_Conditions, y =
## Accident.cleaned$Road_Type): Chi-squared approximation may be incorrect

##
##  Pearson's Chi-squared test
##
## data:  Accident.cleaned$Road_Surface_Conditions and
Accident.cleaned$Road_Type
## X-squared = 196.23, df = 16, p-value < 2.2e-16

CrossTable(x=Accident.cleaned$Road_Surface_Conditions,y=Accident.cleaned$Road
_Type,expected = TRUE,        prop.c = FALSE,        prop.t = FALSE,
prop.chisq = FALSE,        chisq = TRUE,        sresid = TRUE)

## Warning in chisq.test(tab, correct = FALSE, ...): Chi-squared
approximation may
## be incorrect

##    Cell Contents
## |-------------------------|
## |                       N |
```

```
## |              Expected N |
## |           N / Row Total |
## |            Std Residual |
## |-------------------------|
##
##
=============================================================================
===
##              Accident.cleaned$Road_Type
## Ac.$R_S_C    Roundabot   On wy str   Dl crrgwy   Sngl crrg   Slip road
Total
## -----------------------------------------------------------------------------
------
## Dry               6050        2596       14520       67129        1063
91358
##              6034.4      2408.0     14610.5     67250.7      1054.4
##               0.066       0.028       0.159       0.735       0.012
0.723
##               0.201       3.832      -0.749      -0.469       0.265
## -----------------------------------------------------------------------------
------
## Wt or dmp         2189         705        5295       23673         374
32236
##              2129.3       849.7      5155.4     23729.7       372.0
##               0.068       0.022       0.164       0.734       0.012
0.255
##               1.295      -4.963       1.945      -0.368       0.101
## -----------------------------------------------------------------------------
------
## Snow                15           4          81         317           7
424
##                28.0        11.2        67.8       312.1         4.9
##               0.035       0.009       0.191       0.748       0.017
0.003
##              -2.458      -2.146       1.602       0.276       0.952
## -----------------------------------------------------------------------------
------
## Frst or i           94          27         279        1877          15
2292
##               151.4        60.4       366.5      1687.2        26.5
##               0.041       0.012       0.122       0.819       0.007
0.018
##              -4.664      -4.299      -4.573       4.621      -2.227
## -----------------------------------------------------------------------------
------
## Fl o 3. d            2           0          42          61           0
105
##                 6.9         2.8        16.8        77.3         1.2
##               0.019       0.000       0.400       0.581       0.000
0.001
```

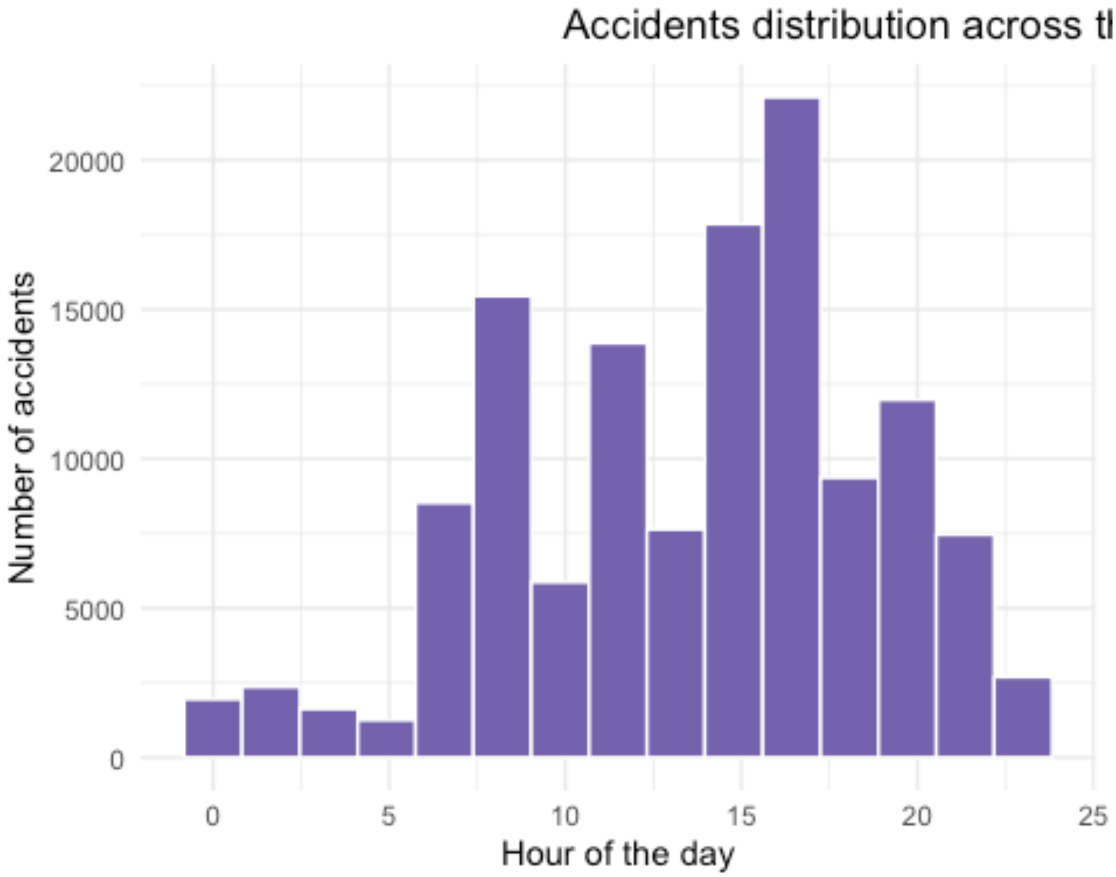
```
##                 -1.874      -1.664       6.152      -1.853      -1.101
## --------------------------------------------------------------------------
------
## Total             8350        3332       20217       93057        1459
126415
##
===========================================================================
===
##
## Statistics for All Table Factors
##
## Pearson's Chi-squared test
## ------------------------------------------------------------
## Chi^2 = 196.2288      d.f. = 16      p <2e-16
```

```
Accident.cleaned<-Accident.cleaned%>%
  mutate(Hour=substr(Time,1,2))
Accident.cleaned$Hour=as.numeric(Accident.cleaned$Hour)
Accident.cleaned%>%
  filter(Hour==17)%>%
  drop_na(Road_Type)%>%
  drop_na(Speed_limit)%>%
  group_by(Hour,Road_Class_1st)%>%
  summarise(n())
```

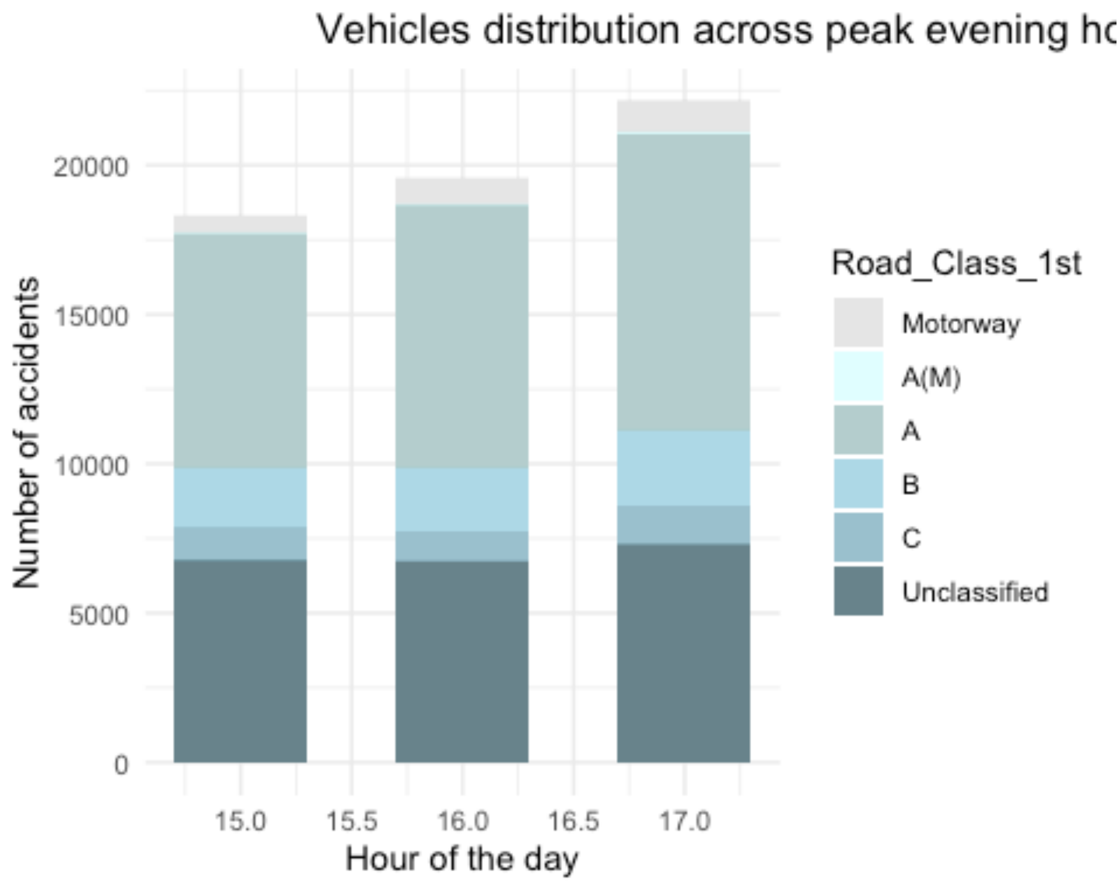
```
## `summarise()` has grouped output by 'Hour'. You can override using the
`.groups` argument.
```

```
## # A tibble: 6 × 3
## # Groups:   Hour [1]
##    Hour Road_Class_1st `n()`
##   <dbl> <fct>          <int>
## 1    17 Motorway         406
## 2    17 A(M)              33
## 3    17 A               4939
## 4    17 B               1361
## 5    17 C                682
## 6    17 Unclassified    4061
```

```
Accident.cleaned%>%
  drop_na(Hour)%>%

ggplot(aes(x=Hour))+geom_histogram(fill="#7463AC",color="white",bins=15)+them
e_minimal()+labs(x="Hour of the day",y="Number of accidents",title = "
Accidents distribution across the day")
```




```
Accident.cleaned%>%
  drop_na(Number_of_Vehicles)%>%
  filter(Hour==15|Hour==16|Hour==17)%>%

ggplot(aes(x=Hour,y=Number_of_Vehicles,fill=Road_Class_1st))+geom_col(width=0
.6)+theme_minimal()+labs(x="Hour of the day",y="Number of accidents",title =
"                Vehicles distribution across peak evening
hours")+scale_fill_manual(values=c("gray90","lightcyan","lightcyan3","lightbl
ue","lightblue3","lightblue4"))
```




```
Accident.cleaned%>%
  drop_na(Number_of_Vehicles)%>%
  filter(Hour==15|Hour==16|Hour==17)%>%
  group_by(Hour,Road_Class_1st)%>%
  summarise(n.veh=sum(Number_of_Vehicles))

## `summarise()` has grouped output by 'Hour'. You can override using the
`.groups` argument.

## # A tibble: 18 × 3
## # Groups:   Hour [3]
##     Hour Road_Class_1st n.veh
##    <dbl> <fct>          <int>
##  1    15 Motorway         553
##  2    15 A(M)              48
##  3    15 A               7829
##  4    15 B               2028
##  5    15 C               1071
##  6    15 Unclassified    6745
##  7    16 Motorway         825
##  8    16 A(M)              72
##  9    16 A               8775
## 10    16 B               2159
```

```
## 11    16 C                  997
## 12    16 Unclassified    6698
## 13    17 Motorway        1040
## 14    17 A(M)              78
## 15    17 A               9896
## 16    17 B               2557
## 17    17 C               1280
## 18    17 Unclassified    7272
```