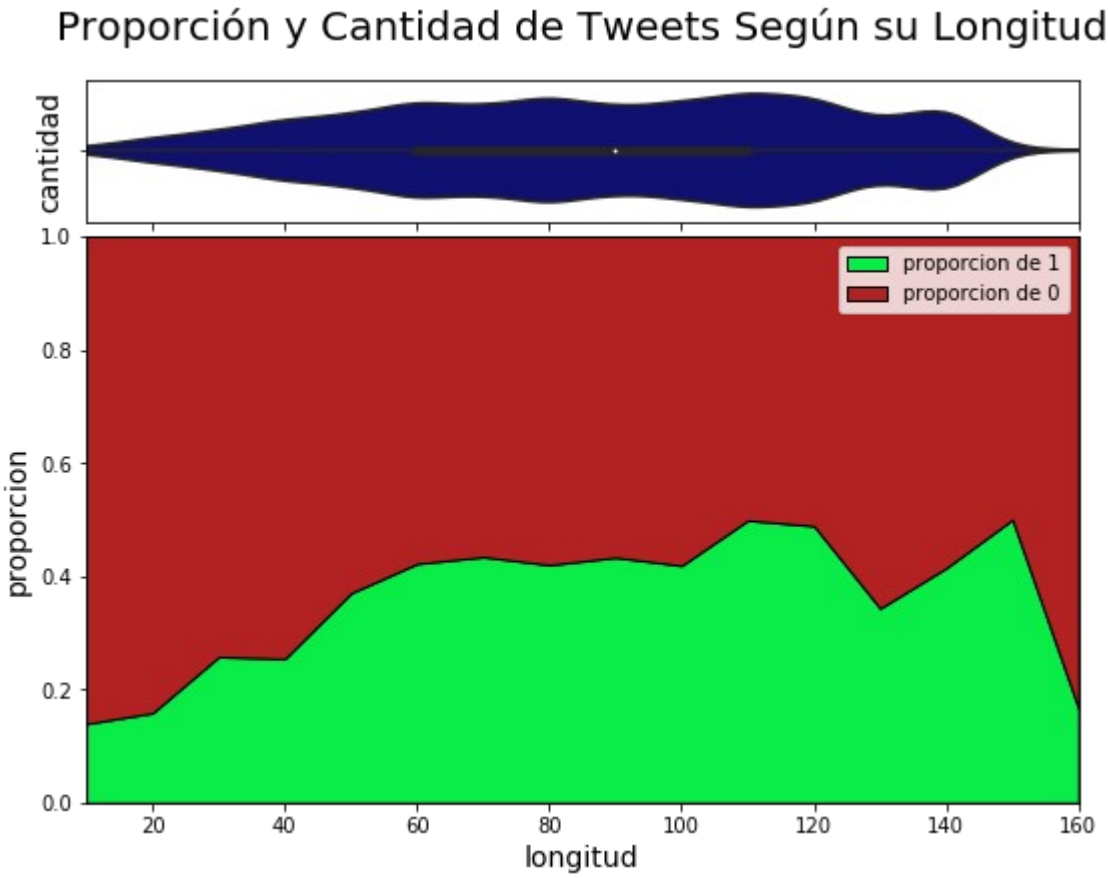
Codigo

```
#seteo para ver los twits completos
pd.set_option('display.max_colwidth', -1)

train = pd.read_csv("train.csv",encoding = "ISO-8859-1")

train.describe()
len(train)

train.sort_values('text').head(15)
##se observan muchos duplicados pero con distinto target, por lo que en lugar de tirar los duplicados, los agrupo, calculo la media del target y luego la redondeo según si su target fue mas veces 1 o 0.

#hay muchos duplicados pero con URL diferentes

#elimino las URLs,que además obstruyen al medir la longitud del tweet
train['text'] = train['text'].replace(r'http\S+', '', regex=True)

#agrupo tweets iguales
train = train.groupby('text').target.median().reset_index()

len(train)
#se redujo casi un 10% la cantidad de twits

train['target'] = train['target'].round(0).astype('int64')
#redondeo a 0 los target que quedaron en 0.5

#ahora calculo la longitud de cada twitt
train['longitud tweet'] = train['text'].str.len()

#la redondeo para agrupar entre múltiplos de 10
train['longitud tweet'] = ((train['longitud tweet']/10).round(0).astype('int64'))*10

#agrupo a partir de los redondeandos
grouped_tweets = train.groupby('longitud tweet').agg({'target':['count','mean']})

#renombro las columnas para quitar el multiindex y agrego la proporcion de 0
grouped_tweets.columns = ['cantidad tweets','proporcion de 1']
grouped_tweets['proporcion de 0'] = 1-grouped_tweets['proporcion de 1']
```

```
#grafico la proporcion de 1 y 0 mediante un stackplot y la cantidad total de twits mediante un
#violin chart compartiendo el eje x para ambos
f, (ax_viol, ax_stack) = plt.subplots(2, sharex=True, gridspec_kw={"height_ratios": (.2, .8)},figsize=(8, 6))
#colores
color_stack = ["#0aec48", "firebrick"]
color_violin = "navy"
#título
ax_viol.set_title("Proporción y Cantidad de Tweets Según su Longitud", size = 20, pad=20)
#valores
x=grouped_tweets.index
y=[ grouped_tweets['proporcion de 1'], grouped_tweets['proporcion de 0'] ]
z=train["longitud tweet"]
# asigno gráficos
sns.violinplot(z, ax=ax_viol, color = color_violin,pad=8.0)
ax_stack.stackplot(x,y,labels=['proporcion de 1','proporcion de 0'],colors = color_stack,edgecolor = "black")
#Label del stackplot
ax_stack.legend(loc='upper right')

#Rango de valores del stackplot
ax_stack.set_xlim(10, 160)
ax_stack.set_ylim(0, 1)

#labels
ax_viol.set_ylabel("cantidad",fontsize=14)
ax_viol.set_xlabel("")
ax_stack.set_xlabel("longitud",fontsize=14)
ax_stack.set_ylabel("proporcion",fontsize=14)

#distancia nula entre los dos graficos
f.tight_layout(pad=0)

plt.show()
```

Gráfico


Proporción y Cantidad de Tweets Según su Longitud
cantidad
proporcion de 1
proporcion de 0
proporcion
longitud
0.0
0.2
0.4
0.6
0.8
1.0
20
40
60
80
100
120
140
160