

Micro-Proyecto 1

Integrantes:

- ✓ Nelson Aldana Código: 201924128
- ✓ Diego Ojeda Código: 201924250
- ✓ Daniel Rojas Código: 201924286

1. Introducción

El abastecimiento de productos alimentarios en las grandes ciudades ha sido un problema de gran impacto en las últimas décadas. Los centros de producción se encuentran normalmente en la periferia de los centros urbanos y usualmente los productores no cuentan con medios propios para llevar sus bienes directamente al consumidor final. Esta conexión entre campo y ciudad, la realizan intermediarios, que a través de varias etapas llevan a cabo la recolección en los centros de producción, el transporte a grandes centros mayoristas de distribución y luego a centros minoristas, donde finalmente, llegan a las manos de los usuarios. Esta serie de etapas entre el productor y el consumidor que lleva a cabo el/los intermedios generan dos fenómenos de interés: los productores no reciben el precio justo por sus cosechas y los consumidores son cargados por el cargo de la cadena de suministro.

Esta *Start Up* agrícola intenta dar solución a este problema, conectando el campo con la ciudad (productores con consumidores), a través de una aplicación que permita al consumidor comprar directamente los productos que el productor pone a disposición a la empresa, la cual se encarga de la distribución a los compradores.

El reto que tiene esta compañía es poder optimizar su sistema de gestión de inventarios, con el fin de minimizar la cantidad (y el costo) de los productos que se pierden por estar demasiado tiempo almacenados, debido a una sobreproducción o una sub-demanda, que se deriva de una desconexión entre la producción y la demanda. Con el fin de cerrar esta brecha, se ha identificado la necesidad de *generar un modelo que pueda predecir la demanda de aquellos productos considerados relevantes con una semana de anticipación*, con el fin de ajustar sus actividades de producción e identificar hábitos de consumo, que permitan optimizar sus operaciones y ofrecer productos de mejor calidad a sus clientes.

2. Metodología Propuesta

Con el fin de construir un modelo satisfactorio, se exploró la base de datos de ventas de la compañía y se encontró el número de productos que maneja la compañía es demasiado elevado (221 productos), lo que llevó a plantear un filtro de productos relevantes. El criterio para definir los productos relevantes fue el dinero que representaba cada producto, es decir, la suma de todos los valores de los pedidos de cada producto.

Se realizó un análisis de Pareto para observar las distribuciones y capturar solamente las variables que representen el 80% del volumen de ventas (figura 1) y que tengan más de 20 pedidos en la línea de tiempo, con el fin de que la predicción se haga sobre los productos cuyas pérdidas económicas sean de mayor impacto. Lo que nos llevó a la selección de **36 productos considerados relevantes** (ver tabla 1).

	Pedido	Precio	cumpercentage
Producto			
VER0016	547	1505800	15.471745
FRU0046	4000	531550	20.933298
VER0051	160	365900	24.692836
FRU0042	110	292980	27.703137
VER0028	1094	271300	30.490682
VER0038	1324	266750	33.231476
VER0010	1211	218500	35.476513
VER0013	858	217800	37.714357
FRU0032	45	207700	39.848427
VER0035	1360	207550	41.980955

Tabla 1: Selección de Variables

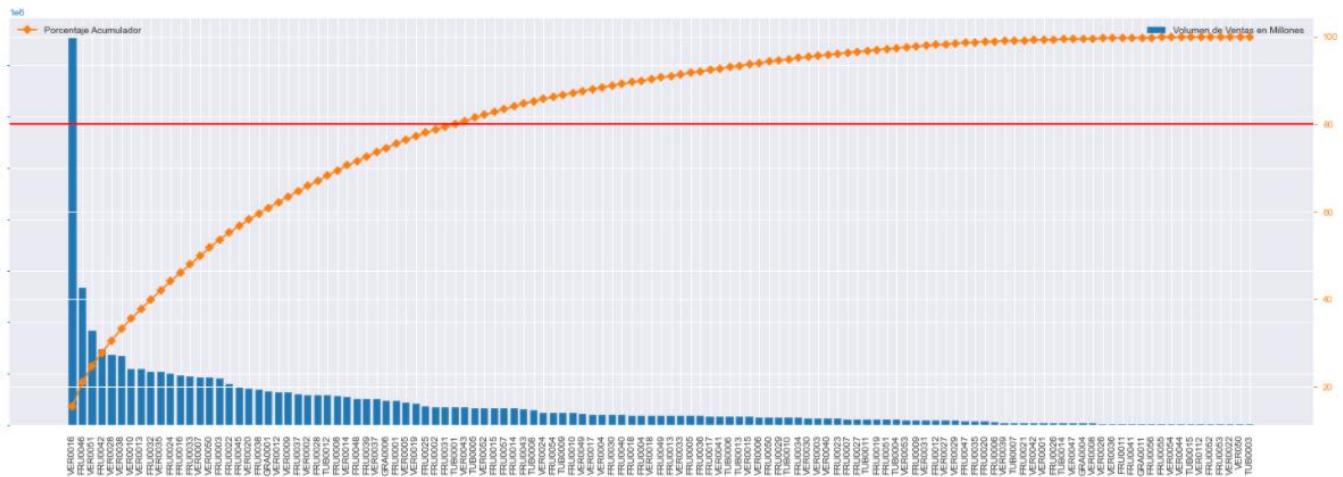


Figura 1: Análisis de Pareto de Selección de Variables

Con el fin de realizar un análisis semanal, se realizó un re-muestreo (*resampling*) semanal y se interpolaron los valores faltantes (ya fuera por ausencia de datos en la base original y por efecto del *resampling*) con el método *time* (figura 2).

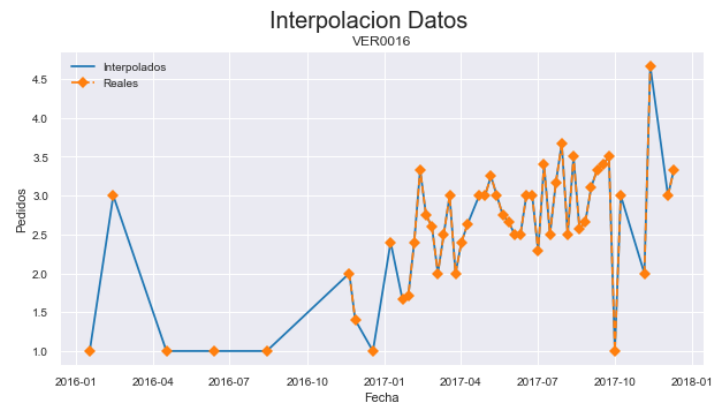


Figura 2: Interpolación de Datos

Se procedió a probar tres tipos de modelos para la predicción de la demanda de cada producto, adaptando cada uno de los modelos (arquitectura, funciones de activación, etc.) buscando el mejor ajuste de cada modelo para cada producto usando como métrica el *MSE*:

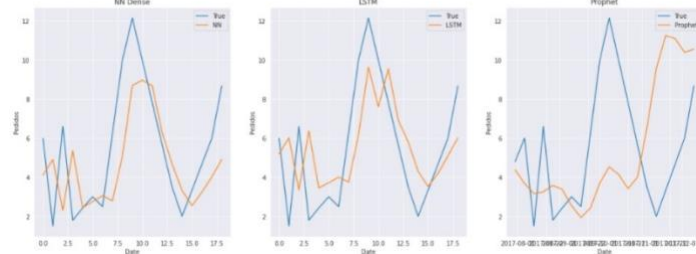
- Una red neuronal densamente conectada.
- Una red de neuronas tipo LSTM
- Un modelo aditivo para predicción (*forecasting*) de series de tiempo llamado *Prophet*.

Posterior a la selección del modelo apropiado para cada producto, se definió una función de predicción de demanda, la cual usa un diccionario de modelos, en la que cada producto tiene asociado el mejor modelo para hacer su predicción (seleccionado del paso anterior). Esta función permite, ingresando el código del producto, obtener la demanda predicha de dada variable de interés para la siguiente semana junto con la presentación visual de la demanda histórica más el resultado.

De manera tal, que por ejemplo para el caso del pepino cohombro se seleccionó un modelo de LSTM con un MSE de 0.065637 y para el caso de la berenjena el mejor modelo es el de una red neuronal densamente conectada con un MSE de 0.90656, como se pueden observar en las figuras (3 y 4) a continuación, con las gráficas comparativas del ajuste de cada modelo a los datos de prueba que se tienen para los dos productos. Cabe aclarar que como se dijo en el párrafo anterior este proceso se repitió para todos los 36 productos que según el análisis de Pareto representaban el 80% de las ventas de la compañía agrícola.

MSE NN: 0.478287
MSE LSTM: 0.065637
MSE Prophet: 13.272483

Pruebas De Modelos Para Pepino Cohombro / Libra



MSE NN: 0.090656
MSE LSTM: 0.093382
MSE Prophet: 12.257413

Pruebas De Modelos Para Berenjena / Libra

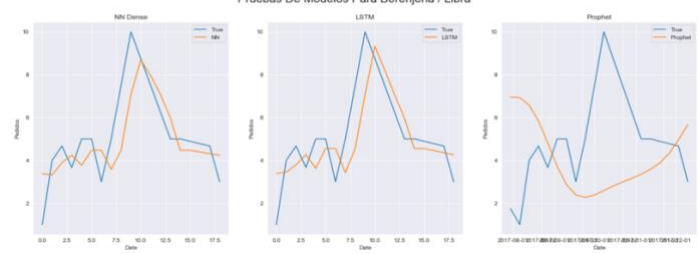


Figura 3: Pruebas para Pepino Cohombro

Figura 4: Pruebas para La Berenjena

3. Resultados

La compañía tendrá la predicción de la demanda de la siguiente semana producto a producto como se observa a continuación, en el caso de la lechuga crespa (modelo de red neuronal, figura 5), se estima que los pedidos serán de 6 unidades, de manera que la *Start Up*, pedirá este tipo de producto en aproximadamente ese número, reduciendo el desperdicio por la predictibilidad del modelo.

Pedido Proxima Semana del Producto
Lechuga Crespa / Unidad
6.304209 Unidades

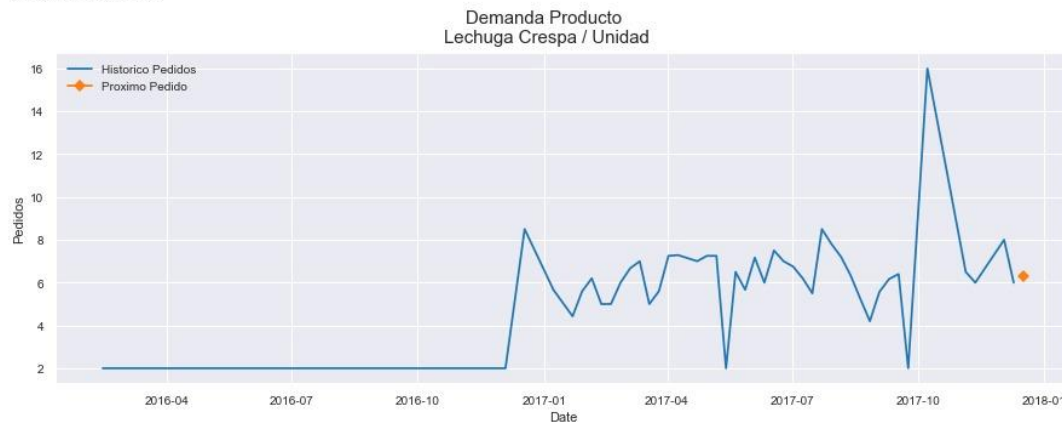


Figura 5: Demanda de Lechuga Crespa

Por su parte, la demanda del producto que más genera ingresos, con una red neuronal, es el champiñón (figura 6), que con un pedido de aproximadamente 3.2 libras, podrá reducir sus desperdicios dado el comportamiento de la demanda de este desde que se tiene data:

Pedido Proxima Semana del Producto
Champiñón / Libra
3.196376 Unidades

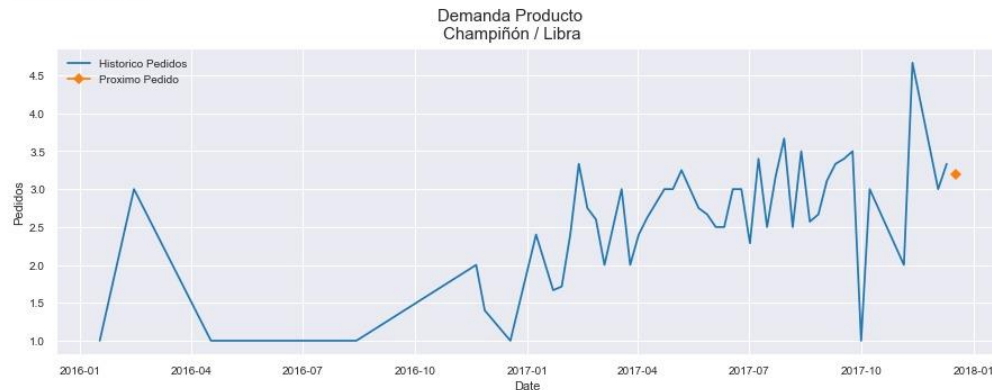


Figura 6: Demanda estimada de Champiñón

4. Conclusiones

La solución entregada a la *Start Up* agrícola logrará reducir el riesgo de desperdicio por sobre inventarios debido a los 32 modelos que predecirán el comportamiento de los productos que significaban el 80% de las ventas de esta. Es de esta manera que el departamento de planeación de esta puede estimar cómo hacer sus órdenes para entregarlas a los consumidores finales y lograr al tener una cadena de abastecimiento óptima. Entregando un modelo que solucione el problema: *generar un modelo que pueda predecir la demanda de aquellos productos considerados relevantes con una semana de anticipación.*

Detrás de esta solución hay un modelo asociado a cada producto, considerando arquitecturas diferentes y haciéndola más flexible de acuerdo también a los cambios estacionales que pueden tener este tipo de productos y los que van a ser más relevantes para las ganancias de la misma. De manera tal que para el estudio puntual los modelos se distribuyeron de la siguiente manera: 18 con redes neuronales densamente conectadas, 18 LSTM y ningún *Prophet*. Dejando así una preferencia por los dos primeros modelos.

El algoritmo de predicción de series de tiempo *Prophet* no arrojó mejores resultados en términos del MSE para ninguno de los productos individuales a los cuales se les aplicaron los modelos, se pudo observar que su ajuste presentaba patrones más suavizados, que no lograban capturar la alta volatilidad de la demanda en ventanas de tiempos pequeñas. Quizá este modelo se pueda desempeñar mejor en series de tiempo con menos valores extremos y cambios abruptos.

5. Referencias:

Browniee, J. Time Series Forecasting With Prophet in Python. Machine Learning Mastery. 2020. Disponible en línea: <https://machinelearningmastery.com/time-series-forecasting-with-prophet-in-python/>

El-Nesr, M. Imputing the Time-Series Using Python. Toward Data Science. 2018. Disponible en línea: <https://medium.com/@drnesr/filling-gaps-of-a-time-series-using-python-d4bfddd8c460>

Franco, C. Notas de Clase: Modelos Avanzados para el Análisis de Datos II. Universidad de los Andes. 2020.