# Prueba Técnica Analista III

Bienvenido a la prueba técnica para Analista III de Riesgos de Mercado y Liquidez del Grupo Bancolombia.

A continuación, se enfrentará usted a un problema muy similar al que sería su día a día en la compañía, por tanto, su desempeño en este conjunto de pruebas será determinante para decidirnos por su perfil para el puesto.

La prueba a continuación descrita medirá su capacidad de programación, análisis, generalización y capacidad de productizar y generar entregables listos para ser usados por los usuarios finales.

## Características de la prueba

La prueba planteada se divide en dos, la primera parte medirá su capacidad para programar (crear scripts) e implementar soluciones acordes con la realidad financiera del producto. La segunda parte medirá su capacidad de generar el producto amigable para el usuario final.

## La segunda parte consiste en la creación de un front-end que interactúe con la función *Calculo\_VaR* que usted creó en la primera parte, recibiendo información de la interfaz y devolviendo el dato de VaR al final.

***Se sugiere al aspirante documentarse sobre “productización” de modelos para la segunda parte de la prueba.***

A continuación, dos links que ilustran la idea:

[Beginners Guide to Model Development](https://heartbeat.fritz.ai/brilliant-beginners-guide-to-model-deployment-133e158f6717)

[Deploying a Python Model with Flask](https://towardsdatascience.com/develop-a-nlp-model-in-python-deploy-it-with-flask-step-by-step-744f3bdd7776" \o "Deploying a Python Model with Flask)

## Prueba de scripting y programación

La prueba es la generación del Valor-en-Riesgo, entendido como la pérdida probable para un percentil , para un portafolio diversificado. Para tal propósito, se dejan las siguientes especificaciones del ejercicio:

* El portafolio por valorar se encuentra en el archivo “Prueba Desarrollo.xlsx”. La fecha de valoración es 08 de abril de 2019, fecha para la cual se calculará un Valor en Riesgo al horizonte (ventana) solicitado. Las condiciones de valoración para ese día se anotan en el archivo adjunto.
* **El entregable es un documento técnico, donde se explique de manera teórica y práctica el ejercicio desarrollado.**
* **También se debe enviar el (los) código(s) realizado(s) en Python que contenga(n) la lectura de la información y el cálculo del VaR según llamados a la función.**
  + El código final debe ser una función denominada ***Calculo\_VaR****,* sus inputs son:
    - Vector de tipo string cuyos valores pueden ser cualquier combinación de País, Institución, Producto. Ejemplo: *[“Colombia”,””,“Renta Variable”]*, *[“Panamá”,””,””]* o *[“”,””,””]*. Default: *[“”,””,””]*.
    - Window: Ventana de tiempo (si no se especifica, el default debe ser 250)
    - Lag: Días con los cuales se calcula la diferencia (default: 10)
    - Alpha: Porcentaje (default: 0.01)
* **Además, un archivo .htm o cualquier tipo de extensión que se abra en Google Chrome, Internet Explorer, Mozilla Firefox con las características planteadas:**
  + Forma de seleccionar la geografía
  + Forma de seleccionar la institución
  + Forma de seleccionar el producto
  + Forma de seleccionar window, lag y alpha.
  + Botón que comience el proceso
  + Impresión del resultado
* Cualquier modificación a esta metodología es permitida mientras exista una justificación, sea operativa o teórica de dicho cambio, junto con la explicación de los efectos de este cambio en el cálculo del VaR.

A continuación, se compartirá con el aspirante algunos conceptos que facilitarán el desarrollo de su prueba:

## Conceptos clave

El cálculo de VaR Simulación histórica se realiza calculando diferencias simuladas (con un parámetro llamado ) a una ventana de tiempo y aplicándolas al valor base de la fecha.

De esta forma, para cada habrá un valor simulado del factor de Riesgo , asignado a cada posición en un vector de tamaño (se pierden algunos datos por la operación).

El será redondeado, con el fin de obtener como Valor-en-Riesgo una posición existente en el vector, no intermedia. Por ejemplo, para una ventana de tiempo de 250, un lag de 10 y un de 1%, el Valor-en-Riesgo es la 3ra peor posición:

Como podrá observar, el archivo “Prueba Desarrollo.xlsx” contiene la información del portafolio ubicado en diferente país, diferente institución del Grupo Bancolombia, diferente tipo de producto. Parte de la evaluación pedirá medir el Valor en Riesgo para subconjuntos de este portafolio: Por país, por Institución, por Producto, asegúrese que su código tenga en cuenta todas las combinaciones de estas posibilidades. Ejemplo: Sólo Producto o Producto por País.

Se sugiere utilizar funciones de Python para calcular los escenarios de simulación .

Puede utilizar cualquier paquete de Python que considere necesario, se sugiere utilizar numpy y pandas, como mínimo.

Puede generar cuántos archivos .py considere necesarios, pero tenga en cuenta que un número exagerado de archivos .py generará ilegibilidad en su código.

Tenga en cuenta que las matrices de factores tienen diferente tamaño, esto no debe suponer ningún problema para usted.

Algunos ejemplos de input para la función:

|  |  |
| --- | --- |
| Llamado | Procedimiento |
| Calculo\_VaR([“Colombia”,””,””]) | Calcula el VaR para Colombia, todas sus instituciones, todos sus productos, obteniendo un único vector de VaR para la suma de los escenarios de Colombia y devolviendo el 3er peor escenario. |
| Calculo\_VaR([“”,””,”Moneda”]) | Sin importar en qué país en qué país está ubicado el portafolio de monedas, se debe obtener escenarios para la suma de éstos, en forma de un único vector y obtener el 3er peor escenario. |

## Calificación de la prueba

Los siguientes criterios serán tenidos en cuenta al momento de calificar la prueba

* Calidad del código: El código es general para cualquier portafolio, es eficiente en términos de uso de funciones, módulos, booleanos y ciclos.
* Calidad y legibilidad de los comentarios: Los comentarios son esenciales cuando de crear código en equipo se trata y facilita la transferencia de conocimiento, por ello, se evaluará la capacidad del aspirante de explicar su código mediante comentarios.
* Calidad de la documentación: La documentación técnica también hace parte esencial del trabajo en el área. Se evaluará una buena documentación que permita entender el proceso implementado.
* A la función **Calculo\_VaR** se le harán los siguientes llamados y se medirá que tan parecido es el output con las respuestas correctas (esto sin usar la interfaz):
  + Calculo\_VaR()
  + Calculo\_VaR([“Colombia”,””,””],300,20)
  + Calculo\_VaR([“”,”Bancolombia Panamá”,“Renta Fija”])
  + Calculo\_VaR([“Colombia”,”Bancolombia”,”Renta Fija”],250,1,0.05)
* Creatividad en la Interfaz de usuario.
* Usabilidad de la interfaz
* Funcionalidad de la interfaz
* *Será bien valorada, pero no necesaria, una buena presentación gráfica de los resultados*

## Entregables

* **El entregable es un documento técnico, donde se explique de manera teórica y práctica el ejercicio desarrollado, en pdf.**
* **También se debe enviar el (los) código(s) realizado(s) en Python que contenga(n) la lectura de la información y el cálculo del VaR según llamados a la función. Archivos .py.**
* **Además, un archivo .htm o cualquier tipo de extensión que se abra en Google Chrome, Internet Explorer, Mozilla Firefox con las características planteadas.**
* **Presentación .pptx o .pdf donde se muestre el procedimiento**